STATISTICAL MODELS FOR LOB SYSTEMS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Gökhan Arıkan

May 2018

STATISTICAL MODELS FOR LOB SYSTEMS

Gökhan Arıkan, Ph.D.

Cornell University 2018

Econophysics is an area of study that aims to understand complex behavior and properties in economic/financial markets. Trades and Quotes (TAQ) data pulled from Johnson Graduate Management School Server from January 2014 was used in this study to analyze intertrade time duration of 39 securities and limit order book (LOB) activity. Discrete Weibull Distribution was shown to be a better model as opposed to Zero Truncated Geometric Distribution to model intertrade time duration after relevant statistical tests. Moreover, the dependance on the market sector and market cap of both shape and scale parameters of Discrete Weibull Distribution were investigated. Finally, LOB activity between two consecutive trades was analyzed for a single security and the Null Hypothesis of poisson distribution was rejected.

BIOGRAPHICAL SKETCH

Gökhan Arıkan was born and raised in Aydın, Türkiye in 1980. He graduated from Bilkent University with Bachelors of Science in Physics in 2003. During his undergraduate studies, he was interested in theoretical condensed matter physics. On the other hand, after he started graduate school at Applied & Engineering Physics of Cornell University, his interests were switched to experimental condensed matter physics. Soon he found himself working on time resolved x-ray studies of thin film deposition, specifically Pulsed Laser Deposition of $SrTiO_3$ thin films. After spending years in x-ray labs, he decided to switch gears and pursue the rest of his graduate studies on a theoretical project. Now, he is finishing his degree on applications of statistical mechanics to Limit Order Book (LOB) Models under supervision of Richard Lovelace and Robert Jarrow as advisors. He is excited to be joining Abu Dhabi Financial Group (ADFG) as a Researcher in Machine Learning in January 2018.

to my family:

Hasip, Cumhur, Mine Nihan, and Merve

and

to beloved Seher.

ACKNOWLEDGEMENTS

I would like to start my acknowledgements with my thesis advisor and special committee members: Richard Lovelace, Robert Jarrow, and Philip Protter. Thank you all for your patience, support, motivation, and giving me freedom about my research topic. Without their support this project would have never come to fruition.

A wise man once said, "The biggest treasure one can be gifted in life are a lovely family and trustable friends". Words cannot express how much I am thankful to my family for their endless support and what they have done for me. Hasip, Cumhur, Mine Nihan, and Merve Arıkan: without your support, I would have never been the person I am right now. During time of ups and downs, it was the unconditional love from my sisters, Mine Nihan and Merve, that helped me to make progress towards my degree. I am also thankful to my uncle Hüseyin Özevcimen for not only teaching me the foundations of mathematics but also piquing my curiosity in physics and astronomy since high school years.

During my Cornell years, I was very lucky to meet and click a special group of people: Wasif Syed, Shaan Qamar, Saad Ahsan, and Hatice Bilici. Thank you all for being my closest friends throughout this journey. English language is not enough to properly describe my feelings about you, however you should know that it was an absolute pleasure sharing every single moment with each of you.

My dearest Arab friends each of them carrying a unique personality also deserve to be acknowledged. I was very honored and felt special to make friendships with Khalid Al Kaabi, Aziz Al Majid, Maan Aldaiel, Hamdan Al Yousefi, Bader Al Monawer, and Mohammed Al Amer. Thank you Khalid for accompanying me in various espresso adventures, thank you Aziz for sharing after midnight insomnia cookies, thank you Maan for enlightening discussions of the

conditional survival probability after a hypothetical zombie apocalypse, thank you Hamdan for sabotaging my diet with kinder chocolates while being roommates, thank you Bader for your energy, adventurous spirit, and unconditional companionship, thank you Mohammed and your lovely wife Anoud for amazing dinners in your place.

Moreover, each of the Turkish friends I made their acquaintance during my studies are also important to me. It would be impossible to mention all the names here however Zeki Durak, Onur Tokel, Oğuzhan Vıcıl, Mehmet Ünlü, Yasin Damgacı, Melik Türker, and Ömer Bilgen would have made the top of the list. Thank you all for valuable times, enjoyable moments, and great discussions all shared together.

Few people I regret meeting so late then quickly became good friends are Haider Syed, Yacine Bourezak, Zehra Sheikh, and Farwah Sheikh. It was a pleasure getting to know each of you and I am looking forward to sharing more moments in the future.

There is one final group of people I have known since my college years. The friendship between us was never depreciated even though the distance has increased over time: Yusuf Danışman, Sinan Esgin, Selim Alayoğlu, Taner Özel, Murat Keçeli, İlker Daştan, and Yunus Zeytuncu. Thank you all for being on my side whenever I need your support.

Finally, I would like to acknowledge the entire Qamar family for their presence, support, and motivation. Renee, Iqbal, Natasha, Shaan, Sonya, and of course Cooper: I love you all.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Physics and Economics

Physics and Economics have very different purviews as fields of knowledge. Physics is interested in the precise description and prediction of physical phenomenon: ranging from quantum phenomenon of atoms to the cosmological underpinnings of the expansion of galaxies. Meanwhile, while economics is a social science interested in the study of different phenomenon entirely: from the incentives of actors in the microscale and how prices are set in international trade on the macroscale. Therefore, at first view, one would think that the scope of physics and economics are different entirely. However, as it turns out, there are bridges between the two disciplines [16, 37, 38].

At the heart of it is the fact that both disciplines are underpinned by rigorous use of mathematics. Both physics and economics seek to develop a predictive and descriptive understanding of global, universal phenomenon from few underlying properties of constituent elements. As a result, several techniques developed in one field find fertile group for application in the latter. This ranges from tools developed in statistical mechanics, i.e. correlation effects, self-organization, self-similarity, scaling, complex systems, equilibrium, and non-equilibrium states. We discuss a few case studies of striking analogies in the field of finance below. The emergence of mathematical finance goes back to early 1900's with Bachelier's thesis which laid the foundation for modern financial mathematics [4]. The physics approach to financial markets should be considered as complementary to the work of mathematical finance and eco-

nomics.

In Physics, Brownian motion was generalized into ideas of fractional brownian motion where memory is required as well as to geometric brownian motion. In Economics, Ito, building up on the works of Bachelier, Wiener and Kologorov invented Ito calculus which is now widely used in the theory of option pricing. In addition, the stochastic calculus of diffusion processes in combination with classical economics were seeded together into arbitrage pricing theory.

Moreover in Physics, the Heat equation is widely known and used. In fact, it found a curious application in the field of finance as well. In particular, diffusion processes continued to be studied in Finance after the development of Ito calculus. They would culminate in the Nobel-prize winning idea of the Black-scholes formula in 1997. This formula is well-known to be the solution to a heat equation (after some required transforms in its system of partial differential equations) [9].

1.2 Econophysics

At the heart of it, econophysics is a hybrid discipline that seeks to apply models and ideas from physics to solve a range of problems in economics and finance [48]. In fact, one would argue that these physics models are not just imported into the discipline; as Mantegna and Stanley [36] state "The word econophysics describes the present attempts of a number of physicists to model financial and economic systems using paradigms and tools borrowed from theoretical and statistical physics." Just like the applications of physics into economics were discussed in Section 1.1 in other areas of economics, for example Brownian motion

in financial economics, econophysics seeks to model economic phenomenon using ideas and tools from condensed matter physics [48].

Econophysics arose in the 1990s through the work of a number of physicists in the statistical mechanics field working in "complexity sciences" [44]. These physicists were not satisfied with explanations and approaches the economists employed as they used simplified theoretical models to agree with empirical data. They applied tools and methods developed by physicists to match financial data sets to explain general economic phenomena [16]. Sudden availability of large financial data was the driving engine in the development of econophysics at this time. The traditional methods proved insufficient in the face of standard economic methods that handled homogenous agents and equilibrium at the time when financial markets were generally dependent on heterogeneous agents and non-equilibrium situations [32]. The term econophysics was coined by Eugene Stanley in 1995 as a term to describe the great amount of papers physicists wrote about the stock and other market problems. Today it is a well-recognized field with courses offered in global universities such as at Leiden University. In fact, Jan Tinbergen, a famous Nobel laureate physicist was awarded the first full professor Econophysics position at Kings College in 2014 [55].

Econophysics's most basic tools are both statistical and probabilistic methods extracted for statistical physics. The kinetic theory of gas has been applied in economics as kinetic exchange models of markets. Others include percolation models, chaotic models (developed in the study of cardiac arrest) and critical self-organizing models and others such as in the earthquake prediction. Attempts have also been made to apply complexity and information mathematical

theories [32]. Economic phenomena result from the interaction of many heterogeneous agents and statistical mechanisms analogy are used to explain the difference in human properties and particles. Statistical mechanism have however been proven to be results of well-established tools as opposed to being used in economic models in potential games. In potential games, it has been established that information via Shannon information based equilibrium is produced as a stochastic dynamical equation.

Several authors and physicists used quantifiers from information theory to assessment the extent or degree of the efficiency of information in stock markets. Zunino et al [56] used in the complexity-entropy Cartesian plane developed as an innovative statistical tool to establish different market efficiency ranking and differentiate dynamics of bond markets. Some authors posit that derivations from the above method is consistent in the case of major company ratings and sovereign instruments [55]. Another study by Bariviera et al [6] found out in an exploration of credits ratings and information efficiency of corporate bonds using Zunino's Cartesian plane that classifications agree with credit ratings.

Econophysics has registered successes in such areas as explanation of "fat tails" in financial data distribution of many kinds as a universal self-similar scaling property. It has applied scale invariant ideas over many data magnitude orders that arise from individual market competitors and aggregates to exploit micro trends systematically and optimally. These fat tails are important mathematically as they embody the risks which are very small and may be neglected [34]. Their employment may prevent this and be made exponentially tiny. It is also applicable in ear change tendency such as its rising and falling of prices with panic reactions of sellers and buyers. These fat tails are also seen

in commodity markets. Fat tails can be obtained by non perturbative methods complication as they have Gaussian approximation derivations such as the black-Scholes theory. Other causes of fat tails include term random number in centre-limit theorem and non econophysics models. It is difficult to test these kinds of models and have as a result received less attention in the traditional analysis of the economy.

Another area of success of econophysics is in the partial equilibrium theory in which phase transition two market models are bounded interdependently with longer investment term agents and speculators who serve to balance the markets fast. The occurrence of market preference is shown. Very little market dynamics are not well explored in traditional economic approaches since these economic systems consist of many agents that interact nonlinearly and therefore exhibit the characteristics of complex systems. Statistical physics and nonlinear dynamics have therefore been proven to be very useful and important in analyzing underlying dynamics of systems.

Economic sciences have not been able to avert such common crises in the world economy such as the credit crunch. The difference between financial mathematics or economics and the physical sciences is the role played by concepts, equations and empirical data. Traditional economics is based on assumptions that become axioms and they include economic agents' rationality, the invisible hand and market efficiency [46]. Physicists have learned to be suspicious of axiom and models. Models that do not work are discarded regardless of whether they are beautiful or mathematically convenient. Physics are finding a foot holds in economics due to their critical nature to their own developed models. These attitudes have contributed to the growth of science but are yet to

succeed in economics which has been reliant on dogmatic ideas.

Clear and large effects have arisen due to reliance on incorrect axiom-based models. The Black-Scholes model for example deems extreme events negligible in the assumption that changes in price have a Gaussian distribution. The use of this model led to the 2008 economic crash when the cash-free mode; destabilized the market. In the 2008 economic crisis, financial products development packages the risk into high yield investments the pricing models were flawed and underestimated the multiple borrowers' probability to default loans. This model therefore again ignored the possibility of a global crisis apart from contributing to its occurrence. There are no mechanisms in traditional economic of understanding wild markets.

In contradistinction, a number of models give ways into the understanding of world markets and its effects. These include the theory of complexity. The solutions still remain undiscovered apart from being fragile to small environmental changes and this renders it irrelevant in the understanding of the happenings. The complexity theory should therefore be used in economic systems and particularly in the financial markets.This implies that classical economics' traditional practices should be done away with to pave way for the development of new tools as proposed by econophysicists [52].

1.3 Complex systems and Financial Markets

A complex system is a collection of many interdependent parts interacting with each other in a collinear collaboration resulting in an emergent self-organizing behavior. There are cases in which problems are hard to understand as a result

their problems hard to find because the causes and effects are not related in any way. The fact that different parts of a complex system are interdependent, the disruption of such processes may have far overarching strange and unforeseeable consequences.

Financial markets have the characteristics of complex system model as its evolution is dictated by the decisions of many traders trying to win in a vast game. The financial markets are dynamic as they evolve and generate great amounts of data on a daily basis. This implies that time is important in the study of financial markets in the face of a dynamic world [46]. Another characteristic of complex systems exhibited by financial markets is the interplay between competition and cooperation. This characteristic is also witnessed in social systems such as ecology groupings, immune systems, economic and social classes, teams, nations, supranational corporations and dotcom ventures.

Complex adaptive systems are a special category of complex systems. These systems are able to change themselves in order to adapt to the changes in the environment. They are also capable of changing the environment to suit them. This open ended system has many heterogeneous agents in a non-linear manner over time with their environment and is also capable of changing their behavior based on experiences. Financial markets also have agents as traders, is furnished by capital in varying amounts and interaction rules which are the commercial laws on the trading arena. Each agent tries to benefit the most by making the biggest profit as they sell and buy financial assets at different times. The complex system in financial markets is characterized by uncorrelated swings of financial indices and crashes as extreme events [55].

The speculative bubbles represent nonlinearity indicators in the financial

markets. The panic that spreads in case of losses and speculative bubbles also a characteristic are examples of autocatalytic nonlinear processes in which small stimuli conditions can cause an extensive imbalance in the system dynamics [43]. This means that there lacks a stable equilibrium as in the financial markets in which when prices go up, agents buy more financial assets and this contributes to further price rise. The same applies to the fall of prices where there is only one force that pulls everything down and no chances for balance for the achievement of a stable price.

The financial markets behaviors also exhibit some characteristics of a complex system. The interaction of investors results in emergent behaviors or aggregation. The rules of trade and investment are the decision rules in the financial markets. Adaptive decision rules are represented by disappearance of anomalies. Agent interaction nonlinear character is shown by the fact that cause and effect are not simplistically linked but may interact to churn out inflated outcomes.

1.4 Tools for Statistical Mechanics

Non equilibrium statistical methods are applied in economics as have been by econophysicists. There are many areas that involve the quasi-thermodynamic processes out of equilibrium. This process takes place with time with rate characteristics. This field works to understand the microscopic level non-equilibrium processes. One tool for statistical mechanics is stochastic method in which the non-equilibrium statistical mechanics incorporate stochastic or random behavior into the system. This behavior destroys information in the en-

semble. Thus is though inaccurate but the randomness is added to ensure that information is concerted into subtle correlations within the system over time or to reflect environment-system correlations. These may seem to be chaotic and pseudorandom on variables. The calculations are made easier by replacing correlations with randomness proper. There are a number of equations in practice such as the Boltzmann transport equation coined from the kinetic theory. It is an important tool in no equilibrium statistical mechanism because of its simplicity. Another tool the BBGKY hierarchy is used in liquids and dense gases and gives a method for Boltzmann-type equations and extends them to include correlations after collisions. The keldysh formalism is a statistical mechanism tool that approaches quantum including stochastic dynamics that re in the keldysh formalism. Near equilibrium methods is another non-equilibrium method dealing with systems perturbed from equilibrium. Their response can be analyzed using linear response theory. Fluctuation-dissipation theorem, Green-Kubo relations, Mori-Zwanzig formalism, Landauer-Buttiker formalism and Onsager reciprocal relations are some of the tools used in this framework to make the fluctuation-dissipation connection. There are also hybrids methods used which are advanced approaches and combines stochastic methods and linear response theory. The Green-Kubo relations can be used with stochastic dephrasing to show integrations between certain agents.

Statistical mechanics forms a branch of theoretical physics embracing probability theory to understand the average behavior of any uncertain mechanical system [49]. Statistical physics have been used to aid in predicting the future trend of the financial markets. Voit [54] also voices his opinion in asserting that financial markets present statistical properties that need to be analyzed using statistical mechanics. Financial markets are analyzed using graphs that

have analytics from statistical physics. The author indicates that market analysts can embrace the Black-Scholes equations to help solve market problems including pricing issues. Voit [54] presents a chapter on scaling financial data, a phenomenon that focuses on the scaling properties of price movements. Voit [54] observes that price movement based on time scales can be compressed into a universal curve through rescaling the time intervals and reading from the created probability densities. Consideration of the Levy-stable probability densities including the specialized Gaussian pdf (probability density function). Interactions of foreign exchange (FX) market data is also built on the probability densities that further make prediction of the forex trends possible. Bury [15] on the other hand indicates that stock markets present a complex system that exhibit a collective phenomenon including their fluctuations, synchronization, non-random structures as well as similarity to market neural networks. Bury [15] shows that data-based methodologies can be applied to allow comparison to statistical physics stimulations of complex systems. The physics statistical simulations are commonly applied in market indices like the Dow Jones and other market-generated indices. Price returns are also generated by the use of big data analytics. Some models include spin glass and agent-based models that aid in giving price fluctuations of markets and determining market behaviors. Some of the common tools and process for statistical analysis include creation of inferred distribution, interpretation of the results and the parameter estimation. Bury [15] uses binary data to set up a model describing the state of market as well as its structure. The tools for predicting market movements can be statistical thermodynamics, and non-equilibrium statistical mechanics.

Physicists are finding a foothold in the financial markets in the last decades and are offering insights and applying their skills of data handling. They are

findings ways to spot a lot of things that conventional economics has not been seeing and can help in times of flash crashes and high-frequency trading. A majority of traders are still using traditional economic modes such as the Gausian model and are unable to handle situations that happen outside these models and are forced to endure hard times due to the dogmatic economic education they hold. The bell curve developed by Carl Gauss has so many statistical models underlying it that serves to explain a number of phenomena in the financial markets. This though leaves some gaps that are being fulfilled by physicists. The territory of econophysics is gaining ground in economics as it combines two disciplines of physics and economics and physics is lately taking over economics and will soon be intertwined fields [34]. There are a number of things that are not answer by traditional financial theories that are being answered by physics techniques as they delve deep into the large quantities of data obtained in the financial markets and models have been developed that previously worked in the field of physical sciences only. Banks and hedge funds have been observed to be luring physicists into their strategy and analysis teams as they are able to coin new theories that will handle the chaos of markets and model complex price derivatives as efficient as those in the field of physical sciences. Economics is swiftly transforming into physics as physicists continue to divine laws that are being used to predict future occurrences despite the complexity in the market segments and relations.

Standard economic theory suggests that agents in markets tend to behave rationally. The observation of extreme large fluctuations in price of financial assets that are not correlated to changes in the fundamental value and the financial bubbles ad crashes however imply that market agents display an irrational behavior.

1.5 Organization of the Thesis

A brief description of the thesis is outlined below. The next chapter, Chapter 2, is going to introduce the concept of the Limit Order Book, as well as define some terminology and present some stylized facts related to it. Chapter 3 will introduce the format of the data source that this work was utilized. It will also discuss the various details and methods used to extract, format, and clean the data so it could be used for analysis. Some statistical facts are also provided in this regard. Chapter 4 and Chapter 5 introduce the primary results section for this thesis. Chapter 4 will talk about distribution of intertrade time duration as well as clustering in trade order size. Meanwhile, Chapter 5 will review some primary results related to the Limit Order Book Activity. Finally, Chapter 6 will end the thesis with a Conclusion and potential future work applying ideas from Machine Learning.

CHAPTER 2

LIMIT ORDER BOOKS

In this chapter, we introduce Limit Order Book (LOB) Models. We are interested in studying Limit Order Book Mechanisms from the context of econophysics. Academically, this interest is motivated in part by the fact that LOB mechanisms can be studied from the perspective of complex system. Indeed, in most LOB systems, the market microstructure is one with multiple players in a very dynamic system; this begs itself to be described through the lens of a complex system. We argue that rules for universality, scaling, and emergences ought to be applied. Second, there is significant practical interest in understanding LOB mechanisms from the perspective of optimum portfolio liquidation problems [2, 7, 29, 40]. Modeling and analysis of LOB systems are useful, or even mandatory, for approaching current problems in the literature such as optimum order execution, market impact minimization and efficient trading algorithm formulations [20, 21]. After introducing LOB systems in this chapter, we discuss methods and results from empirical studies on historical LOB data for the rest of the dissertation.

2.1 What's a Limit Order Book?

There are two broad types of markets that provide liquidity for market exchanges. In the quote-driven market, market makers set bid and ask quotes for orders. In an order-driven market, there are no market makers, except maybe brokers, and prices are achieved via a decentralized system of many players that are interacting, responding to and shaping the historical order book. Because of

this, any adequate model of how prices arise in the order-driven market is intractable; one must explain the interaction between a large number of random players who can arrive at any time and change their orders whenever they want [11, 22]. The order book is simply the sum of all existing and active limit orders. There are primarily two types of limit orders. A buy order, also called a bid, is an order to buy if the price of the commodity falls below a specified price whereas a sell limit order, also called an offer (or ask), is an order to sell if the price of the commodity rises above a specified price. The market prices are set via the ask price which is the lowest offer and the bid price which is the highest bid. Trade happens when opposite orders at same price level match. These days order-driven markets account for more than half of global stock exchanges with a LOB at the center of the trading including the NYSE, NASDAQ and LSE [31, 45]. There are a few very basic observations one can make about the microscopic structure of the LOB system.

2.2 Terminology

In this sub-section, we take some time to define some commonly used terms when discussing LOB models [18, 28].

Definition 2.1. Bid Price: The highest stated price amongst active buy offers at a given time.

Definition 2.2. Ask Price: The lowest stated price amongst active sell offers at a given time.

Definition 2.3. Relative Bid Price: This is the difference between the bid price and a given price.

Definition 2.4. Relative Ask Price: This is the difference between a given price and the ask price.

Definition 2.5. Bid-Ask Spread: This is the difference between the ask price and the bid price.

Definition 2.6. Mid Price: This is the middle price between the ask and bid price.

2.3 Stylized Facts

The results of more than half a century of time series data of order-driven markets has shown that despite nave imaginations that different assets from different markets ought to display different properties, there are actually some statistical properties that are common to all these assets from different markets and across different time points [16, 18]. These statistical regularities across different assets, markets and times are known as stylized facts. Because these properties are common denominators amongst a broad cross-section of markets, one gains generality but loses precision in describing such assets. Nonetheless, the stylized facts do impose a lot of constraints on any proposed model for this stochastic process so that there is a lot of intuition to be gained from them [16, 18]. In this sub-section, we introduce a selection of stylized facts summarized from [18].

Distribution of returns and log-returns are fat-tailed: Unconditional returns of assets display a fat-tailed behavior that seem to obey power-law. In particular, the tail index seems to be finite and normally between two and five for most data sets. This excludes stable laws with infinite variance and normal distributions. Still the precise form is hard to determine. Even after we take

a look at conditional returns corrected via volatility clustering, the returns still show a fat-tailed behavior (although less fat).

There is no correlation between returns: Linear autocorrelation of asset returns decay very quickly within intraday time spans (in about 20 mins). Beyond this, there is no significant correlation.

Volatility clustering: Different measures of volatility show high degrees of positive autocorrelation in time-scales of days. Spikes in volatility seem to cluster together in time.

Aggregational normality: As one increases the time scale over which returns are calculated, the returns seem to re-distribute into a gaussian distribution. Changing the time scales over which the returns are calculated changes the normal distribution.

Autocorrelation and long memory of returns: Absolute returns exhibit slow decay in autocorrelations with time lags that exhibit power law with decay exponent of about [0.2,0.4] which indicates some memory and time-dependence.

CHAPTER 3

EMPIRICAL DATA

3.1 Trades and Quotes Data

Trades and Quotes (TAQ) data contains intraday transactions data for all securities listed on all exchanges. For the purpose of this research, TAQ data from January 2014 was extracted from Johnson Graduate Management School (JGSM) servers. There are five different files stored for each trade day: date file, (trades/quotes) index files, and (trades/quotes) binary files. The date file contains the trading dates as well as start/end position in each index files for each trading date. Index files contain the transaction (quote) date and its start/end position for each security. Trade binary file contains trade time, trade price, trade size, exchange on which the trade occurred for each security. Quotes binary file contains quote time, bid price, offer price, bid size, offer size, exchange on which the quote occurred for each security. In order to be more specific, **ct01a.idx** represents the trade index file for the first trading day of the month January while **cq01c.bin** represents the quote binary file for the third trading day of the month January. When we consider all trading days from a year (252 trading days) the size of the yearly data is approximately 5 TeraBytes.

3.2 Data Preparation

In order to maximize the efficiency in reading data from the binary files, first SAS was used to read the index files and export the trade (quote) details (date,

start position, end position) as a csv file for each trade (quote) date. Then these files were read, merged, and subsetted in R for the list of securities of interest and saved as a csv file. This final index file including the trade details of securities of interest for all trade dates was used to read trade binary file using SAS and to extract a csv file for each security and each trade date. Finally, the trade files were read in R and converted to RData format to make the analysis faster using highfrequency package. Similar procedure for the securities of interests was followed for quotes index and binary files as well.

When reading and extracting data, the interplay of SAS and R is necessary for practical purposes because both languages have advantage on one another. For example, while R is superb at vector or matrix operations and has fast built in data structures, SAS is great at extracting data from binary files.

Raw data is not ready for the analysis and it requires cleaning and preprocessing because of the various errors and bad records. Although we had the option to filter the data given a particular exchange (such as NASDAQ), we chose to skip this option and include data from all exchanges. Then, the trades and quotes happening outside of the exchange hours (9:30am to 4:00pm) were excluded. Finally, the observations having zero prices (quotes) and abnormal sale conditions were deleted. All the cleaning procedure was achieved by built-in tools in R's highfrequency package which made data processing quite convenient.

Moreover, trade direction is inferred via Lee-Ready rule/algorithm using R's highfrequency package as well [35].

3.3 Exchanges

Order-driven markets (with a LOB) account for more than half of all stock exchanges in the world. A security can be traded on any exchange on which it is listed; however, to be listed it must pay a fee and satisfy the requirements of the stock exchange in question which typically dictate minimums with regards to the number of outstanding stock, total valuation and total income over the last several years. While dual listing, that is being listed in more than one exchange is possible, few companies typically do so. We list some basic background and listing requirements of some famous stock exchanges that we studied below:

NASDAQ : The NASDAQ is an American stock exchange and is the second-largest stock exchange in term of market capitalization. The listing requirements are that all companies have at least 1.25 million shares at 70$ million valuation and an income of more than 11$ million over three years.

NYSE : The NYSE is an American stock exchange and is the largest stock exchange in terms of market capitalization. The listing requirements are that all companies have at least 1 million shares valued at above 100$ million and an income of more than 10$ million over three years.

In addition, AMEX, Boston, Chicago, Pacific, and Philadelphia are also among the exchanges that were used in this work.

Symbol	Name	Market Cap	Industry	Sub Sector
MSFT	Microsoft Corporation	608.05B	Technology	Computer Software
INTC	Intel Corporation	189.98B	Technology	Semiconductors
IBM	International Business Machines Corporation	150.08B	Technology	Computer Manufacturing
TWTR	Twitter, Inc.	13.19B	Technology	Computer Software
AMD	Advanced Micro Devices, Inc.	9.56B	Technology	Semiconductors
GRPN	Groupon, Inc.	2.67B	Technology	Advertising
HSTM	Healthstream Inc.	763.37M	Technology	Computer Software
AMZN	Amazon.com, Inc.	472.17B	Consumer Services	Catalog/Specialty Distribution
WMT	Wal-Mart Stores, Inc.	262.69B	Consumer Services	Department Specialty/Retail Stores
EXPE	Expedia, Inc.	18.58B	Consumer Services	Transportation Services
CAR	Avis Budget Group, Inc.	3.4B	Consumer Services	Rental/Leasing Companies
YELP	Yelp, Inc.	3.77B	Consumer Services	Other Consumer Services
LUB	Luby's Inc	70.18M	Consumer Services	Restaurants
UNP	Union Pacific Corporation	92.1B	Transportation	Railroads
FDX	FedEx Corporation	60.46B	Transportation	Air Freight/Delivery Services
AAL	American Airlines Group, Inc	22.89B	Transportation	Air Freight/Delivery Services
JBLU	JetBlue Airways Corporation	6.23B	Transportation	Air Freight/Delivery Services
BRS	Bristow Group, Inc.	320.07M	Transportation	Transportation Services
PG	Procter and Gamble	220.44B	Basic Industries	Package Goods/Cosmetics
ALB	Albemarle Corporation	15.72B	Basic Industries	Major Chemicals
MTX	Minerals Technologies Inc	2.52B	Basic Industries	Major Chemicals
OMN	Omnova Solutions, Inc.	509.24M	Basic Industries	Specialty Chemicals
T	AT & T Inc	206B	Public Utilities	Telecommunications Equipment
AWK	American Water Works	15.55B	Public Utilities	Water Work
POR	Portland General Electric Company	4.21B	Public Utilities	Electric Utilities: Central
CLNE	Clean Energy Fuels	363.7M	Public Utilies	Natural Gas Distribution
CLRO	ClearOne, Inc	61.14M	Public Utilies	Telecommunications Equipment
JNJ	Johson & Johnson	375.71B	Healthcare	Major Pharmaceuticals
AET	Aetna Inc	57.05B	Healthcare	Medical Specialities
SGEN	Seattle Genetics, Inc	8.62B	Healthcare	Biotechnology: Biological Products
ACHN	Achilion Pharmaceuticals, Inc.	533.37M	Healthcare	Major Pharmaceuticals
CEMI	Chembio Diagnostics, Inc.	73.85M	Healthcare	Major Pharmaceuticals
XOM	Exxon Mobil Corporation	352.15B	Energy	Integrated Oil Companies
COP	Conoco Phillips	62.29B	Energy	Integrated Oil Companies
FANG	Diamondback Energy, Inc.	10.06B	Energy	Oil & Gas Production
ALDW	Alon USA Partners, LP	797.87M	Energy	Integrated Oil Companies
ROYT	Pacific Coast Oil Trust	62.12M	Energy	Oil & Gas Production

Table 3.1: Securities of Interest

3.4 Securities of Interest

The securities used for the purpose of research were chosen from different industries with variety of market caps. Table 3.1 summarizes the details of the securities listed in descending order of market cap within each industry.

CHAPTER 4

RESULTS I: DISTRIBUTION OF INTERTRADE TIME DURATION

4.1 Introduction

In financial markets, not only the returns of a security but also the waiting time between two consecutive trades is considered to be a random variable. Previous work done revealed anomalous waiting times as well as scaling patterns of intertrade time duration [30, 41, 47]. However the past research was limited to trades data from one particular stock exchange, NYSE specifically, and relatively outdated data from 1999. Given the rise of highfrequency trading as well as the ability to place orders across multiple exchanges, we feel the necessity to revisit the same problem encompassing all stock exchanges with recent trades data from January 2014 [19].

The main purpose of this chapter is to investigate the statistical distribution of waiting time between two consecutive trades in other words intertrade time duration. Trade activity for 39 securities from diverse sectors and all exchanges were recorded. Then the trade data is filtered and preprocessed as described in Chapter 3. One of the limiting factors however is the time resolution of the Trades and Quotes (TAQ) data. All events happening within one second are recorded as one second in the binary data hence for the purpose of analysis, TAQ data was split into two parts. The trade activity happening less than one second ($\Delta t < 1s$) were discarded and else ($\Delta t \geq 1s$) were kept for further analysis [8, 14].

The previous research done mainly explored exponential and weibull dis-

tributions [30, 41, 47] which are both continuous distributions. On the other hand our data is discrete in time therefore we investigated discrete versions of exponential and weibull distributions which are geometric and discrete weibull respectively. Moreover, hypothesis testing was achieved via likelihood ratio test and confidence intervals were found via Maximum Likelihood Estimation (MLE) [41]. In the final section, exploration of conditional variables of intertrade time duration such as buy/sell asymmetry of the orders, trade size, trade time of the day, sector and market cap of the security, and exchange were investigated.

Before going into details of the analysis and results, the Table 4.1 below summarizes some of the key statistics about the securities of interest where where N is the total number of trades, V is the total volume in millions, $\langle \Delta t \rangle$ is the mean intertrade time duration, $\langle \Delta \rangle$ is the mean trade size over 21 trading days in January 2014. The statistics is restricted to the part of the data where $\Delta t \geq 1s$.

Our first result is going to be about order size clustering.

4.2 Order Size Clustering

Table 4.2 describes how executed trades are distributed at different sizes, Δ, split by two categories of the intertrade time duration (Δt) for each security. An analysis reveals a majority of trade executions occur at size $\Delta = 100$ for both fast $\Delta t < 1$ and slow $\Delta t \geq 1$ time scales of intertrade time duration. Order size clustering was also previously mentioned in [8, 17].

In addition, scatter plots of the percentages of the trades happening at size $\Delta = 100$ as a function of the logarithm of total number of trades, N, for both

Symbol	N	V(1e6)	$\langle \Delta t(s) \rangle$	$\langle \Delta \rangle$	Min Price ($)	Max Price ($)
AAL	164949	55.991	2.98	339.45	25.06	34.19
ACHN	38097	14.775	12.89	387.83	3.22	4.36
AET	54386	5.501	9.03	101.15	67.18	72.12
ALB	20553	1.693	23.86	82.35	62.39	67.29
ALDW	5417	0.434	90.39	80.19	13.75	16.9
AMD	47994	21.568	10.23	449.38	3.35	4.6
AMZN	113761	13.122	4.32	115.34	357.84	407.94
AWK	24849	1.957	19.76	78.75	41.16	42.68
BRS	13992	1.393	35.02	99.55	70.16	77.41
CAR	50364	8.335	9.75	165.5	36.68	42.48
CEMI	873	0.39	494.67	446.78	3.35	3.75
CLNE	36053	7.554	13.63	209.52	11.32	12.85
CLRO	1182	0.26	396.78	220.05	8.62	9.77
COP	100588	8.759	4.88	87.08	64.36	70.92
CVLT	44759	6.295	10.97	140.65	63	77.51
EMKR	5579	1.231	83.13	220.59	4.71	5.32
EXPE	68731	9.819	7.15	142.86	64.19	72.19
FANG	37133	5.305	13.22	142.88	44.05	53.26
FDX	67794	5.382	7.24	79.39	131.08	144.39
GRPN	131340	59.139	3.74	450.27	9.9	12.42
HSTM	10716	1.494	45.74	139.43	26.7	34.64
IBM	142248	10.307	3.45	72.45	175.35	190.81
INTC	250362	97.836	1.96	390.78	24.41	26.98
JBLU	82234	31.466	5.97	382.64	8.45	9.45
JNJ	164151	14.039	2.99	85.53	88.16	95.37
LUB	952	0.094	505.82	98.83	6.51	7.69
MSFT	263760	93.955	1.86	356.21	34.63	37.88
MTX	10518	0.814	46.4	77.38	51.31	60.03
OMN	4245	0.405	115.2	95.44	8.63	10.11
PG	147226	15.592	3.34	105.91	75.28	81.7
POR	14594	1.409	33.64	96.57	29.13	30.39
ROYT	3796	0.616	128.73	162.26	12.5	14.04
SGEN	37647	5.641	13.05	149.83	38.35	49.45
T	135113	24.327	3.64	180.05	32.01	35.28
TWTR	170508	21.952	2.88	128.74	55.6	70.43
VECO	27383	3.633	17.91	132.66	31.81	38.05
WMT	128522	12.153	3.82	94.56	73.65	81.26
XOM	170285	16.231	2.89	95.31	91.71	101.22
YELP	71471	6.576	6.87	92.01	66.49	83.96

Table 4.1: Statistics for the Securities

	$\Delta t < 1$				$\Delta t \geq 1$			
Symbol	$\Delta < 10$	$10 \leq \Delta < 100$	$\Delta = 100$	$100 < \Delta \leq 1000$	$\Delta < 10$	$10 \leq \Delta < 100$	$\Delta = 100$	$100 < \Delta \leq 1000$
AAL	0.65 %	3.47 %	61.51 %	31.4 %	0.5 %	2.33 %	60.62 %	32.92 %
ACHN	0.65 %	4.01 %	58.8 %	32.7 %	0.46 %	3.27 %	63.03 %	26.39 %
AET	8.96 %	22.33 %	57.37 %	11.18 %	12.25 %	22.51 %	56.48 %	8.67 %
ALB	13.15 %	21.79 %	56.3 %	8.69 %	15.79 %	21.67 %	56.24 %	6.26 %
ALDW	11.14 %	33.59 %	43.14 %	11.72 %	22.48 %	26.45 %	43.23 %	7.64 %
AMD	10.35 %	7.63 %	33.12 %	36.15 %	4.09 %	19.68 %	44.19 %	25.03 %
AMZN	11.22 %	26.21 %	50.9 %	11.45 %	8.42 %	18.18 %	58.37 %	14.73 %
AWK	10.69 %	23.61 %	53.73 %	11.74 %	15.88 %	31.66 %	45.97 %	6.41 %
BRS	13.07 %	28.31 %	51.33 %	7.22 %	13.07 %	22.21 %	60.44 %	4.21 %
CAR	2.86 %	7.55 %	77.88 %	11.46 %	3 %	5.34 %	78.65 %	12.55 %
CEMI	0 %	3.3 %	46.03 %	44.81 %	0.34 %	3.44 %	40.78 %	46.85 %
CLNE	1.78 %	9.04 %	66.06 %	21.8 %	1.02 %	5 %	66.36 %	25.87 %
CLRO	1.74 %	8.4 %	55.34 %	32.33 %	0.76 %	7.53 %	59.73 %	29.44 %
COP	6.64 %	17.9 %	59.9 %	15.05 %	11.87 %	35.98 %	43.23 %	8.73 %
CVLT	4.08 %	10.35 %	75.27 %	9.98 %	3.19 %	6.16 %	80.59 %	9.53 %
EMKR	0.93 %	5.22 %	67.08 %	24.52 %	1.6 %	7.73 %	72.13 %	16.97 %
EXPE	8.72 %	12.49 %	68.35 %	10.11 %	5.57 %	8.13 %	74 %	11.82 %
FANG	2.81 %	15.38 %	70.87 %	10.65 %	1.56 %	8.26 %	80.14 %	9.58 %
FDX	9.51 %	28.79 %	49.94 %	11.64 %	15.91 %	31.26 %	44.98 %	7.8 %
GRPN	1.72 %	2.14 %	51.97 %	39.45 %	0.73 %	1.03 %	60.86 %	30.67 %
HSTM	8 %	24.08 %	59.07 %	8.32 %	5.79 %	12.94 %	73.32 %	7.36 %
IBM	10.95 %	30.06 %	48.2 %	10.55 %	14.66 %	43.08 %	36.63 %	5.5 %
INTC	1.86 %	3.39 %	49.93 %	40.53 %	0.6 %	2.31 %	53 %	39.69 %
JBLU	2.82 %	2.71 %	51.46 %	39.2 %	0.94 %	1.41 %	63.09 %	29.81 %
JNJ	6.72 %	20.33 %	55.19 %	17.07 %	15.55 %	38.31 %	36.71 %	9.16 %
LUB	8.92 %	31.85 %	39.49 %	18.9 %	15.44 %	30.99 %	40.23 %	12.82 %
MSFT	2.28 %	4.26 %	49.92 %	40.6 %	0.92 %	3.16 %	52.83 %	39.17 %
MTX	15.33 %	31.86 %	47.14 %	5.66 %	15.55 %	22.44 %	58.53 %	3.47 %
OMN	4.9 %	22.19 %	56.29 %	16.41 %	7.89 %	22.97 %	59.03 %	9.99 %
PG	6.59 %	18.39 %	56.98 %	17.18 %	15.03 %	36.63 %	38.13 %	9.8 %
POR	10.25 %	24.92 %	50.69 %	13.87 %	12.29 %	20.23 %	55.16 %	12.22 %
ROYT	12.96 %	27.31 %	39.5 %	18.92 %	14.23 %	30.66 %	43.36 %	10.93 %
SGEN	3.16 %	11.49 %	73.13 %	11.86 %	2.53 %	7.91 %	76.08 %	12.97 %
T	4.3 %	13.04 %	41.95 %	37.69 %	11.02 %	42.56 %	26.79 %	17.93 %
TWTR	3.77 %	25.07 %	47.82 %	21.26 %	7.89 %	41.21 %	36.88 %	13 %
VECO	4.31 %	11.46 %	72.42 %	11.55 %	3.43 %	7.37 %	77.75 %	11.08 %
WMT	6.31 %	20.04 %	55.76 %	17.12 %	13.55 %	30.87 %	45.14 %	10.16 %
XOM	6.21 %	20.38 %	55.8 %	17.02 %	12.29 %	34.04 %	42.57 %	10.78 %
YELP	8.88 %	31.82 %	46.62 %	12.25 %	8.84 %	33.43 %	49.32 %	8.15 %

Table 4.2: Order Size Percentages

fast ($\Delta t < 1s$) and slow ($\Delta t \geq 1s$) time scales are shown in Figures 4.1 and 4.2 respectively. We can observe that for the fast time scales and securities with market cap greater than 100B, more than half of the trades are happening at size $\Delta = 100$ for the majority of the securities.

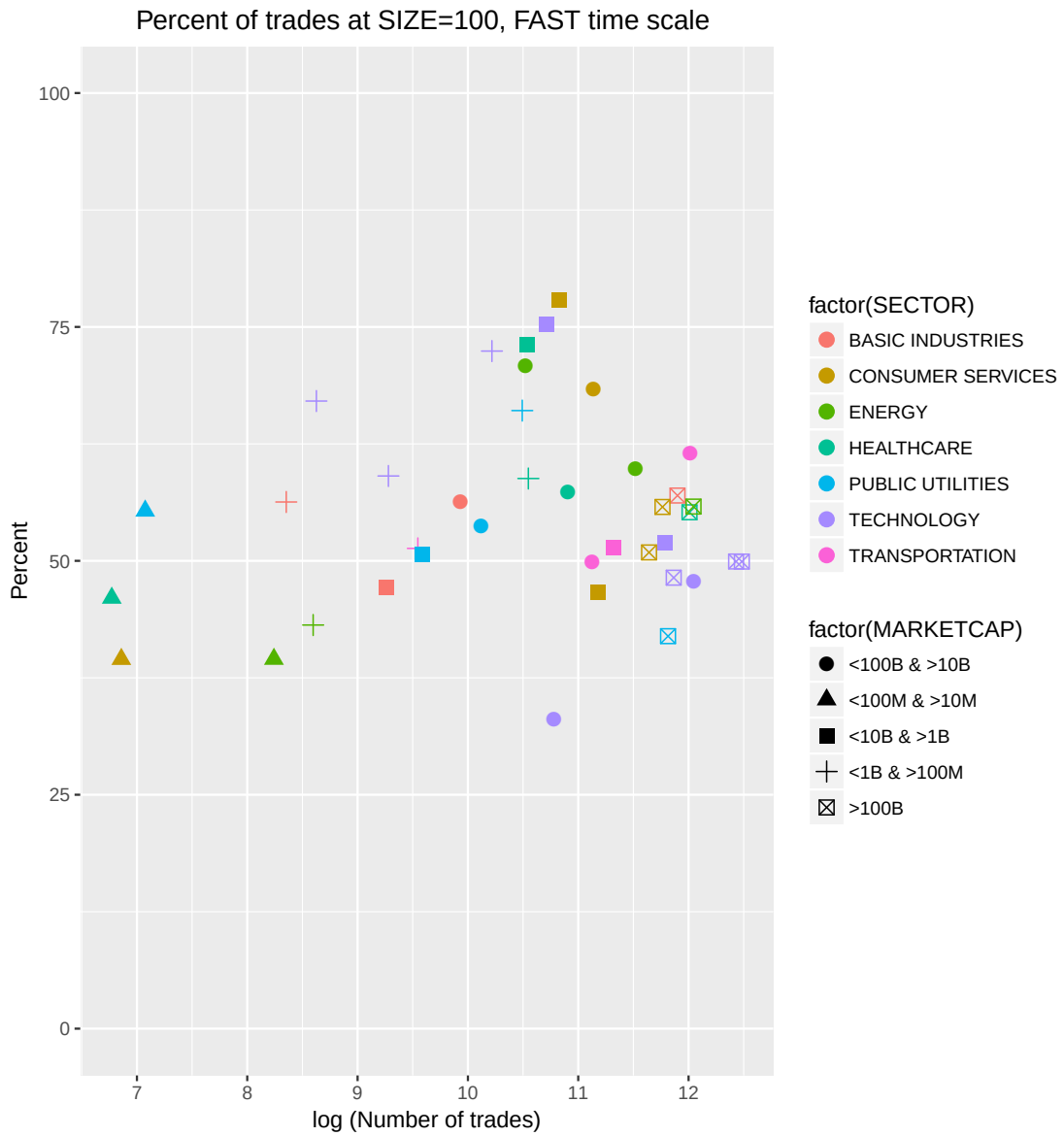

Figure 4.1: Percent of the trades at $\Delta = 100$ & $\Delta t < 1$

4.3 Intertrade Time Duration

The histogram for intertrade time duration of the securities IBM as well as PG, AMZN, CAR, and HSTM for the trade times up to 1 minute ($\Delta t \in [1, 60]$) are shown in Fig 4.3 and Fig 4.4 respectively. Fig 4.3 plots a histogram for the number of trades. Trade distributions for the above mentioned securities are shown

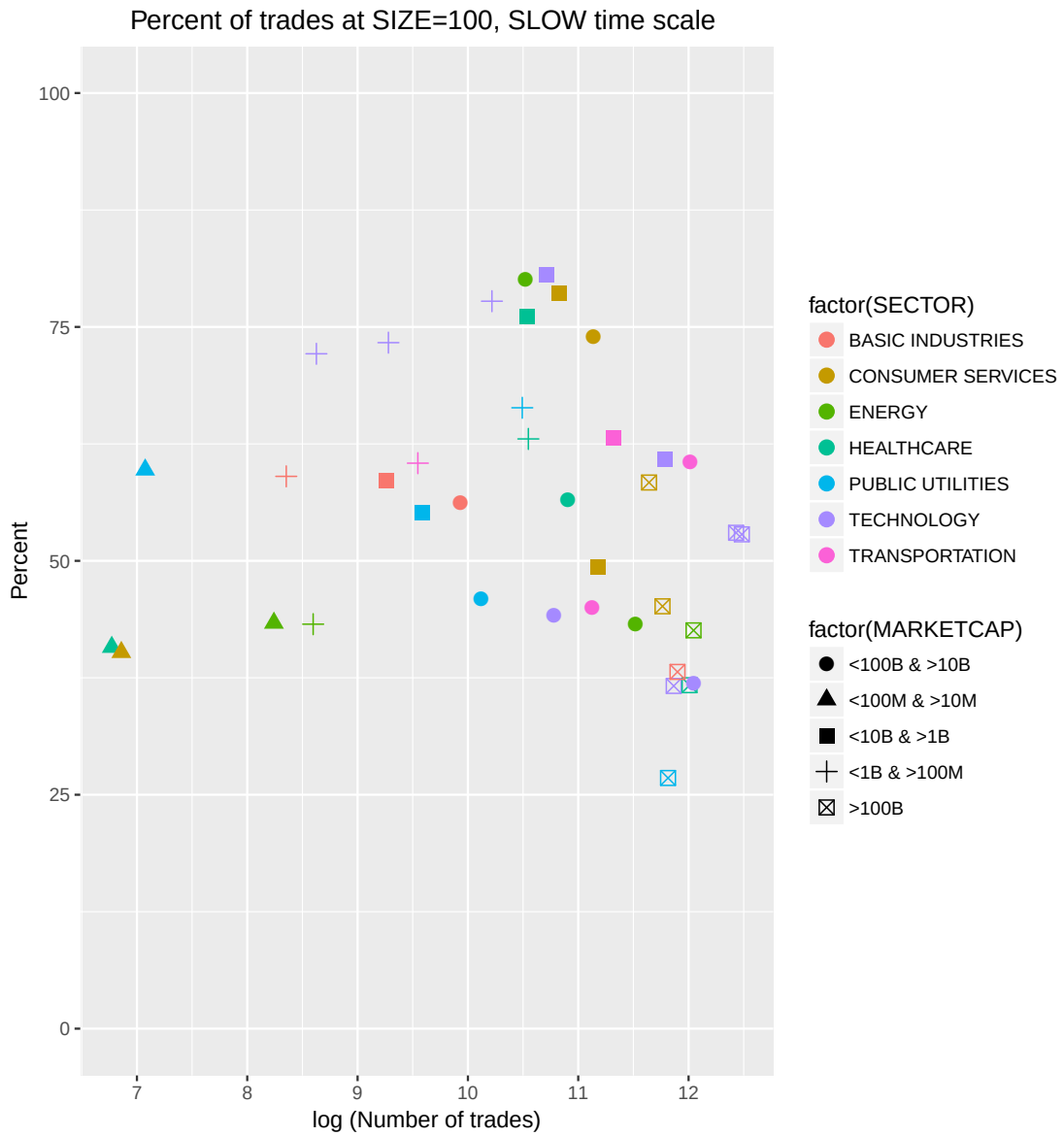

Figure 4.2: Percent of the trades at $\Delta = 100$ & $\Delta t \geq 1$

in Fig. 4.4. This section examines the suitability of approximating these trade times with a zero-truncated geometric distribution, over a discrete weibull distribution.

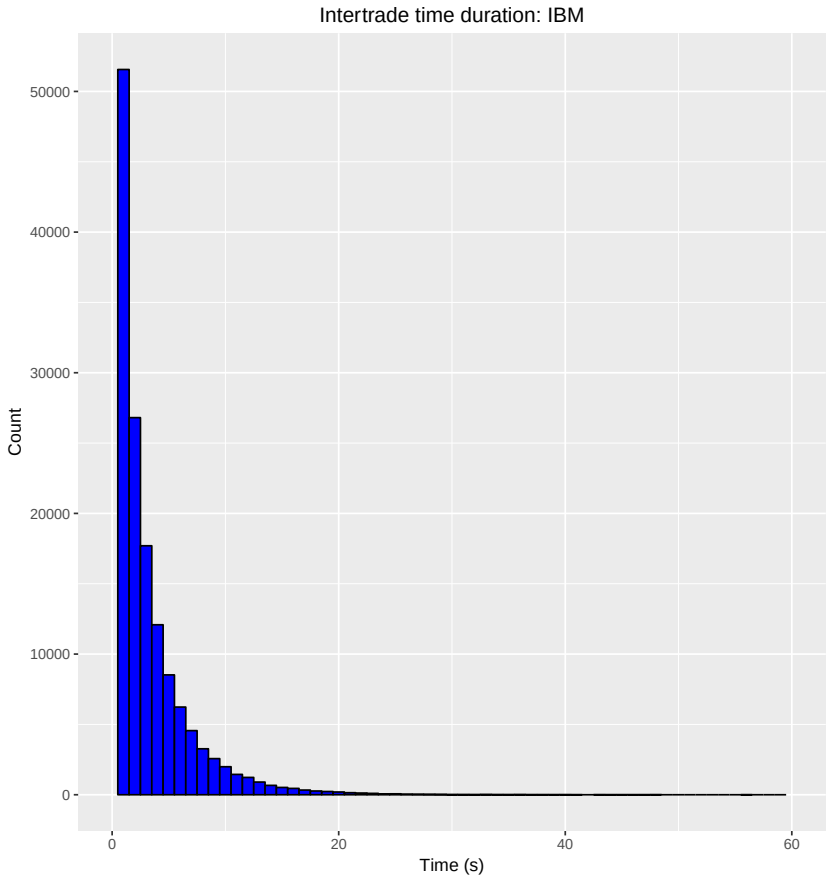

Figure 4.3: Intertrade time duration histogram for IBM

4.3.1 Zero Truncated Geometric Distribution

Zero truncated geometric distributions point to certain interpretations when they are used to describe phenomenon. In particular, they capture the idea that a given number of events occur in a fixed interval in time and that they occur with a known constant probability and independent of the time since the last event. In the context of intertrade time duration, a zero truncated geometric distribution would model perfectly random behavior where the probability of trades occurring at any given time or given any history is constant.

In other words, if at each second $t \geq 1$, the probability of trade execution is

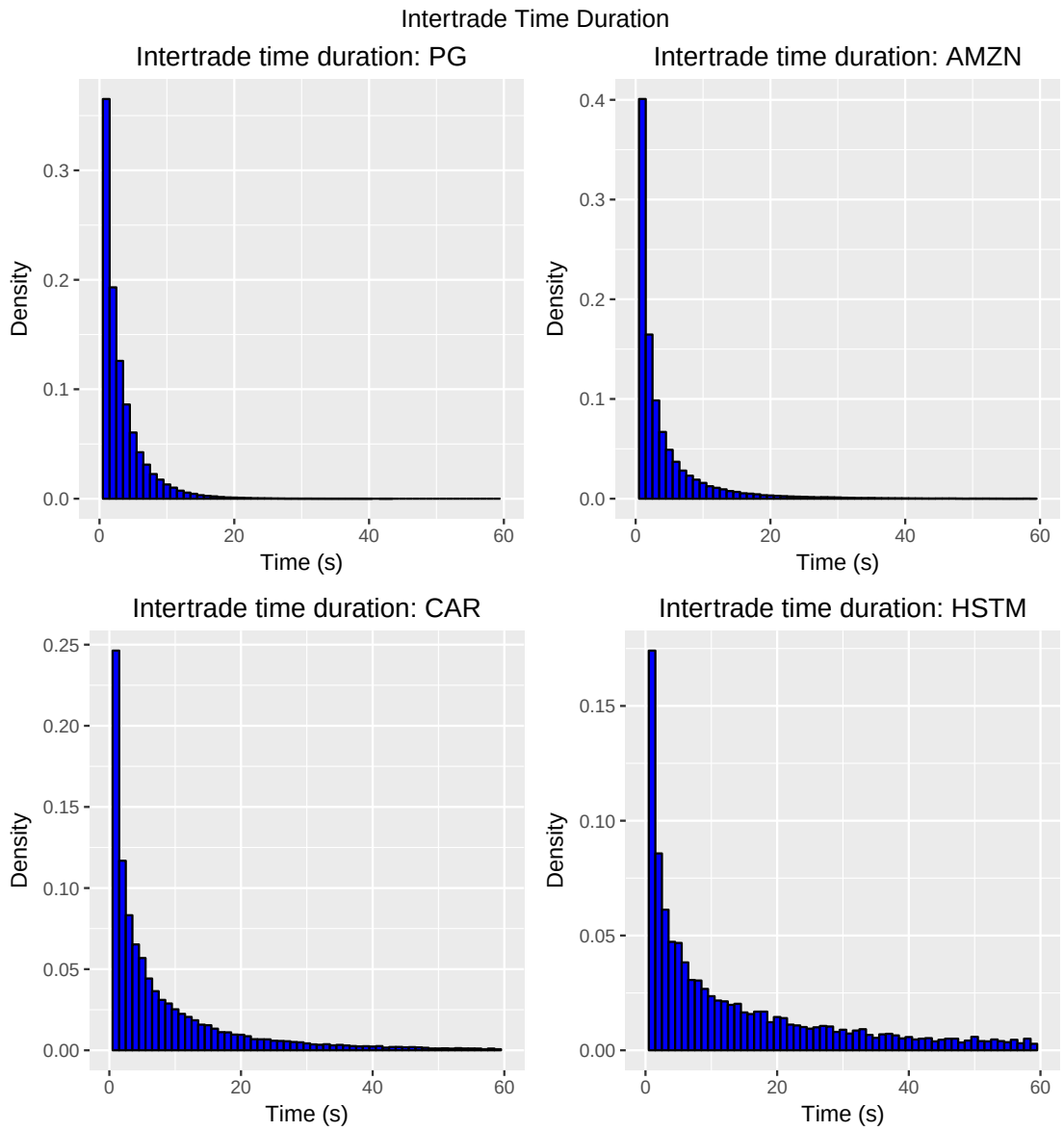

Figure 4.4: Intertrade time duration histogram (normalized scale) for PG, AMZN, CAR, and HSTM

identically p, then the distribution for the number of seconds X until the trade executes is given by the probability mass function of a Zero Truncated Geometric Distribution:

$$f(x) = p(1-p)^{x-1},\ x = 1, 2, 3, \ldots \tag{4.1}$$

Here, the expected number of seconds until trade execution is given by $1/p$, with complementary cumulative distribution $P(X > \tau) = (1-p)^{\tau}$.

4.3.2 Discrete Weibull Distribution

Weibull distributions on the other hand can have multiple interpretations. When shape parameter is equal to 1, they reduce to zero-truncated geometric distributions. However, if the shape parameter is less than 1, they model processes that have a dependence with the time since the last event. In fact, the probability of an incident is higher as we decrease the time since the last event. In another words, multiple events are more likely to occur together.

The probability mass function of Discrete Weibull Distribution is given as:

$$f(x) = q^{x^{\beta}} - q^{(x+1)^{\beta}} \tag{4.2}$$

where q is the scale parameter and β is the shape parameter. When $\beta = 1$, a quick rearrangement of the terms and redefining the starting value of x reveal that:

$$
\begin{aligned}
f(x) &= q^x - q^{(x+1)} \\
&= q^x(1-q) \qquad [\text{let } p = (1-q)] \\
&= p(1-p)^x \qquad \text{when } x = 0, 1, 2, \ldots \\
&= p(1-p)^{x-1} \qquad \text{when } x = 1, 2, 3, \ldots
\end{aligned}
\tag{4.3}
$$

Hence, the Zero Truncated Geometric Distribution is a special case of Discrete Weibull Distribution. This fact is utilized later to quantify evidence for/against the simpler model in Equation 4.1.

4.3.3 Intertrade Time Duration Fits

Maximum likelihood parameter estimates from fitting candidate distributions to the observed intertrade time duration data are provided in Table 4.3. Fig 4.5 and Fig 4.6 overlay the fitted densities on the normalized histograms, for IBM, PG, AMZN, CAR and HSTM securities. Evidently, the discrete weibull distribution is a better model for the intertrade time duration data, an observation which is formalized in the following section.

4.4 Hypothesis Testing

Model fits for each candidate model for security intertrade time data were obtained via maximum likelihood estimation as described in Sec 4.3.3. In this section, model selection on the distribution for security intertrade times is conducted via hypothesis testing.

	D. Weibull Params		Z.T. Geometric Param.
	q	β	p
AAL	0.5208	0.6895	0.3357
ACHN	0.8074	0.7213	0.0917
AET	0.7967	0.7595	0.1136
ALB	0.8846	0.7911	0.0604
ALDW	0.8666	0.6879	0.047
AMD	0.8001	0.7532	0.1088
AMZN	0.5934	0.6503	0.2324
AWK	0.9111	0.874	0.0601
BRS	0.909	0.8363	0.0537
CAR	0.7606	0.6843	0.1101
CEMI	0.9102	0.7819	0.0423
CLNE	0.8378	0.754	0.0819
CLRO	0.8121	0.61	0.0535
COP	0.7291	0.8372	0.2047
CVLT	0.7308	0.6387	0.1105
EMKR	0.9004	0.7626	0.0443
EXPE	0.6969	0.6566	0.1446
FANG	0.7957	0.6985	0.0913
FDX	0.774	0.7781	0.1391
GRPN	0.6137	0.7406	0.2676
HSTM	0.8576	0.7236	0.0604
IBM	0.6361	0.8248	0.2895
INTC	0.4074	0.761	0.5095
JBLU	0.6872	0.6889	0.1698
JNJ	0.6079	0.8578	0.3341
LUB	0.9079	0.8029	0.0478
MSFT	0.3919	0.7804	0.5368
MTX	0.91	0.8241	0.0505
OMN	0.9126	0.7994	0.044
PG	0.6336	0.8404	0.2996
POR	0.9218	0.8588	0.049
ROYT	0.9422	0.8849	0.0384
SGEN	0.7501	0.6296	0.0942
T	0.6473	0.818	0.275
TWTR	0.5109	0.6905	0.347
VECO	0.8096	0.6876	0.0785
WMT	0.6858	0.8711	0.2616
XOM	0.5869	0.8385	0.3465
YELP	0.7484	0.7472	0.1468

Table 4.3: Discrete Weibull and Zero Truncated Geometric Fit Parameters

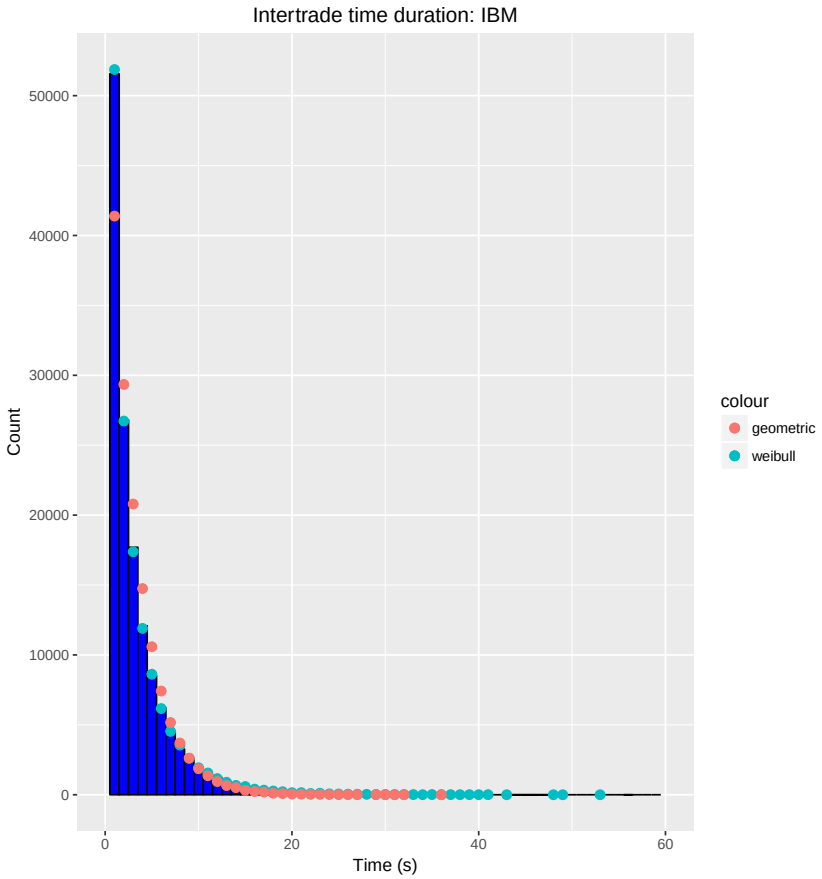

Figure 4.5: Fitting Δt for IBM

4.4.1 Likelihood Ratio Test

The classical setting for simple hypothesis testing sets up a statistical comparison between the following assertions: $H_0 : \theta \in \Theta_0$ vs. $H_A : \theta \in \Theta_1$. In both cases, the distribution of the data is fully specified $X \sim f(x|\theta)$ and one needs to accept H_0 or reject it in favor of H_1 based on evidence provided in the data.

If $T(X)$ is a test statistic, then the hypothesis test resolves into creating a *decision rule* for how large $T(X)$ needs to be provide enough evidence against H_0; in particular, the goal is to determine a critical value c, such that we will "reject H_0 when $T(X) \geq c$". Naturally, any number rules may be constructed

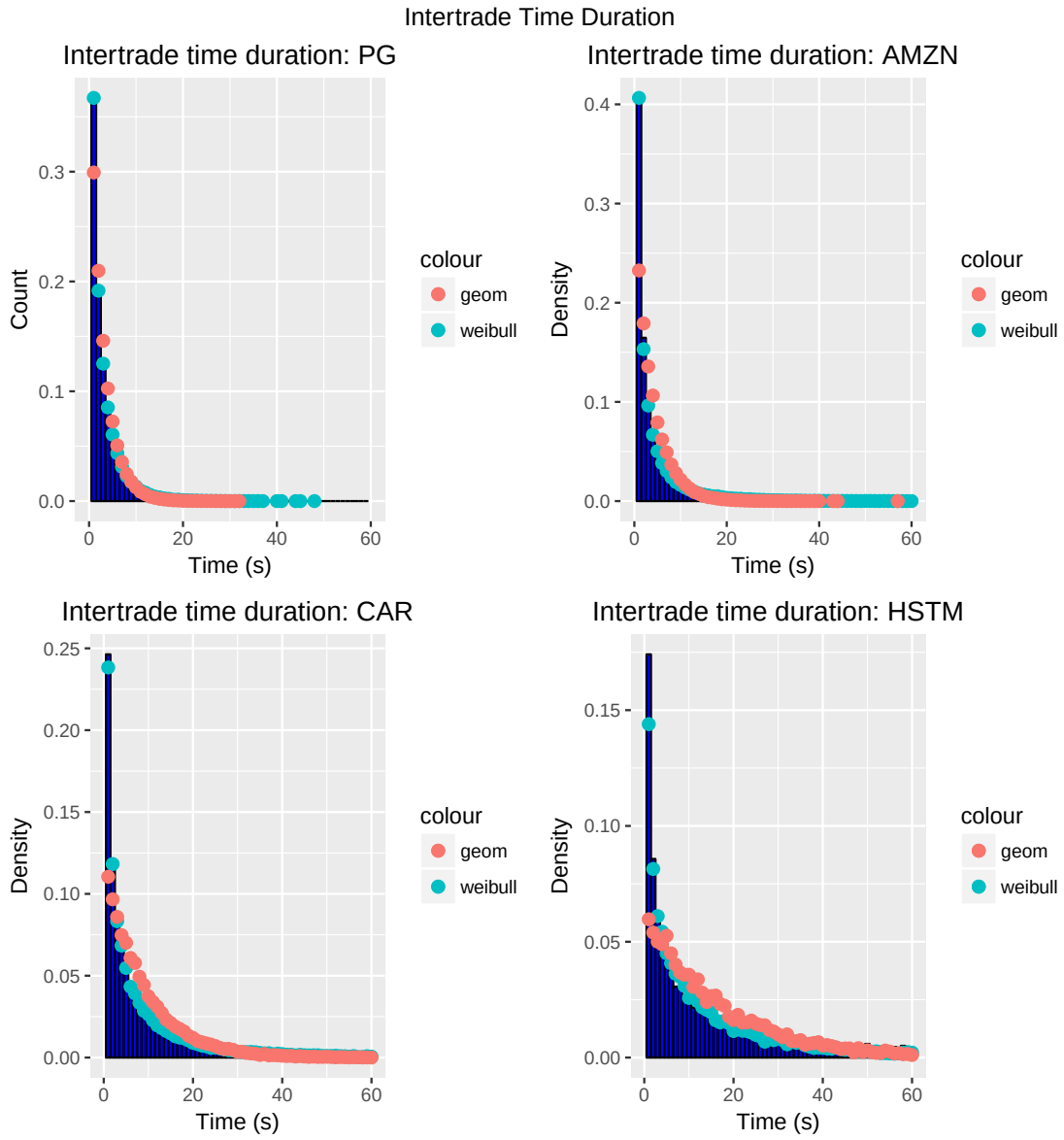

Figure 4.6: Fitting Δt for PG, AMZN, CAR, and HSTM

using various critical values, and one also has complete freedom in choosing the test statistic, T, which may be any functional of the data.

The Likelihood ratio test (LRT) defines the decision rule for this test as

$$\text{reject } H_0 \text{ if } T = \Lambda(X) = \frac{f(x|\theta_1)}{f(x|\theta_0)} \geq c. \tag{4.4}$$

For any $c > 0$, the Neyman-Pearson lemma states that LRT is the *most powerful*

test at a significance threshold $\alpha = \alpha(c) = P(\Lambda(X) \geq c|H_0)$. The latter is simply the false-positive rate or Type-I error (i.e., the chance of reject H_0 under LRT when H_0 is true), which one typically controls to a desired level of precision.

LRT requires nested models, namely models in which the simpler model can be obtained by reduction of the more complex model by imposing constraints on the parameter space, Θ. When the size of the data is large, one commonly appeals to Wilks' theorem, which provides an asymptotic distribution for $T(X)$ under LRT (of course, in simple settings it may be possible to obtained an exact form for the distribution of the test-statistic in which case, that is preferred). In particular,

$$\begin{aligned} H_0 &= \text{Zero Truncated Geometric Distribution} \\ H_1 &= \text{Discrete Weibull Distribution} \\ D &= 2\ln T(X) = 2\,(\text{loglik for } H_1 - \text{loglik for } H_0) \sim \chi^2(k), \end{aligned} \tag{4.5}$$

where $k = |\Theta_1| - |\Theta_0|$, or the difference in the degrees of freedom between the complex and simplified models. Given the test statistic, and its closed-form distribution, we may calculate the p-value for the hypothesis test given the observed data,

$$p = P(\chi^2(1) > c),\ c = F^{-1}_{\chi^2(k)}(1-\alpha). \tag{4.6}$$

Finally, if $p < \alpha \iff D(x) > c = F^{-1}_{\chi^2(k)}(1-\alpha)$, we reject H_0 in favor for H_1.

4.4.2 Confidence Intervals using MLE

Asymptotic normality of MLE provides an approximate confidence interval for model parameters in the large data setting, becoming increasingly accurate as $n \to \infty$. Under regularity conditions [24, 27, 42], one has

$$\sqrt{n}(\hat{\theta} - \theta_0) \to N\left(0, I^{-1}(\theta_0)\right), \tag{4.7}$$

where $\hat{\theta}$ is the MLE estimate, θ_0 is the true unknown parameter of the distribution and $I^{-1}(\theta_0)$ is inverse Fisher Information matrix. When the model is correctly specified, one also has $-1/n\nabla^2 \ln L(\hat{\theta}) \xrightarrow{p} I(\theta_0)$, and hence an approximate 95% confidence interval for θ_0 is given by

$$\hat{\theta} \pm 1.96\sqrt{\frac{-\nabla^2 \ln L(\hat{\theta})}{n}}. \tag{4.8}$$

4.4.3 Results

Section 4.3.2 demonstrated that the zero-truncated Geometric distribution was a special case of the discrete Weibull distribution, with $\theta = (q, \beta = 1)$. Hence, the distribution of the LRT test statistic is approximately $\chi^2(2-1)$.

For all securities in our trades data, the p-value is calculated and results summarized in Table 4.4. For all securities, $p \approx 0$, indicating that intertrade time durations are better modeled by the discrete Weibull Distribution. Note: though multiple independent hypothesis tests are conducted, because the p-value is so small, even a Bonferroni correction applied to control the family-wise error to $\alpha = 0.05$ by choosing $\tilde{\alpha} = 0.05/\{\#\text{tests conducted}\}$. The decision to favor the discrete Weibull distribution for all securities remains the same.

Finally, confidence intervals for the shape-parameter β for the discrete Weibull distribution are computed as prescribed in Section 4.4.2 and summarized in Table 4.4.

	95% **CI**		**Log-likelihood Ratio Test**	
	β_l	β_u	D(x) = $2\ln T$	p-value
AAL	0.686	0.693	27198.38839	0
ACHN	0.715	0.7276	6417.632629	0
AET	0.754	0.765	6484.418428	0
ALB	0.7815	0.8007	1610.659078	0
ALDW	0.6667	0.709	671.6017926	0
AMD	0.7473	0.759	5946.269588	0
AMZN	0.6467	0.654	29897.62098	0
AWK	0.8647	0.8832	661.4679165	0
BRS	0.8236	0.8489	582.8518825	0
CAR	0.679	0.6896	11498.16216	0
CEMI	0.7202	0.8437	41.28317801	0
CLNE	0.7472	0.7607	4449.796258	0
CLRO	0.5618	0.6581	189.2137977	0
COP	0.8326	0.8418	4466.965947	0
CVLT	0.6334	0.6441	14365.04259	0
EMKR	0.7412	0.784	404.587441	0
EXPE	0.6522	0.6611	19048.49085	0
FANG	0.6922	0.7047	7544.666844	0
FDX	0.773	0.7832	6616.18903	0
GRPN	0.7368	0.7445	15913.17605	0
HSTM	0.7105	0.7366	1438.096543	0
IBM	0.8207	0.8288	6685.486631	0
INTC	0.7574	0.7645	16019.84778	0
JBLU	0.6846	0.6932	17313.16307	0
JNJ	0.8538	0.8618	4553.838833	0
LUB	0.7337	0.8721	27.46531309	0
MSFT	0.7768	0.7841	13138.89022	0
MTX	0.8091	0.8391	472.3873638	0
OMN	0.7721	0.8267	182.0317411	0
PG	0.8364	0.8445	5534.114899	0
POR	0.846	0.8716	432.2956765	0
ROYT	0.8521	0.9178	43.7419486	0
SGEN	0.6238	0.6354	12503.02767	0
T	0.8139	0.8221	6995.131396	0
TWTR	0.6871	0.6939	27882.00008	0
VECO	0.6804	0.6949	5892.533882	0
WMT	0.8668	0.8754	3203.253348	0
XOM	0.8346	0.8424	6196.220655	0
YELP	0.7424	0.752	9447.787567	0

Table 4.4: Discrete Weibull Fit shape parameter, β, Confidence Intervals and LRT results for all securities.

4.5 Conditional Variables

In this section, we examine the parameters of the fits to the Weibull distribution against conditional variables such as sector and market cap of the various securities in question.

4.5.1 Market/Industry

The major industries of the securities were chosen are: Basic Industries, Consumer Services, Energy, Healthcare, Public Utilities, Technology, and Transportation [25]. The data was split into 7 different categories given their sector. The shape and scale parameters of the Discrete Weibull Distribution for securites from different sectors/industries are shown in Fig 4.7 and Fig 4.8. Ultimately, we did not observe any effect from industry on either q, the scale parameter, or β the shape parameter. However, the scale parameter is inversely related to the number of trades.

4.5.2 Market Cap

The data was split into 4 different categories given their market cap. The shape and scale parameters of the Discrete Weibull Distribution for securites from different market caps are shown in Fig 4.9 and Fig 4.10. Ultimately, we do see a clustering effect on the scale parameter, q, with regards to the market cap. On the other hand, this is not seen in the shape, β.

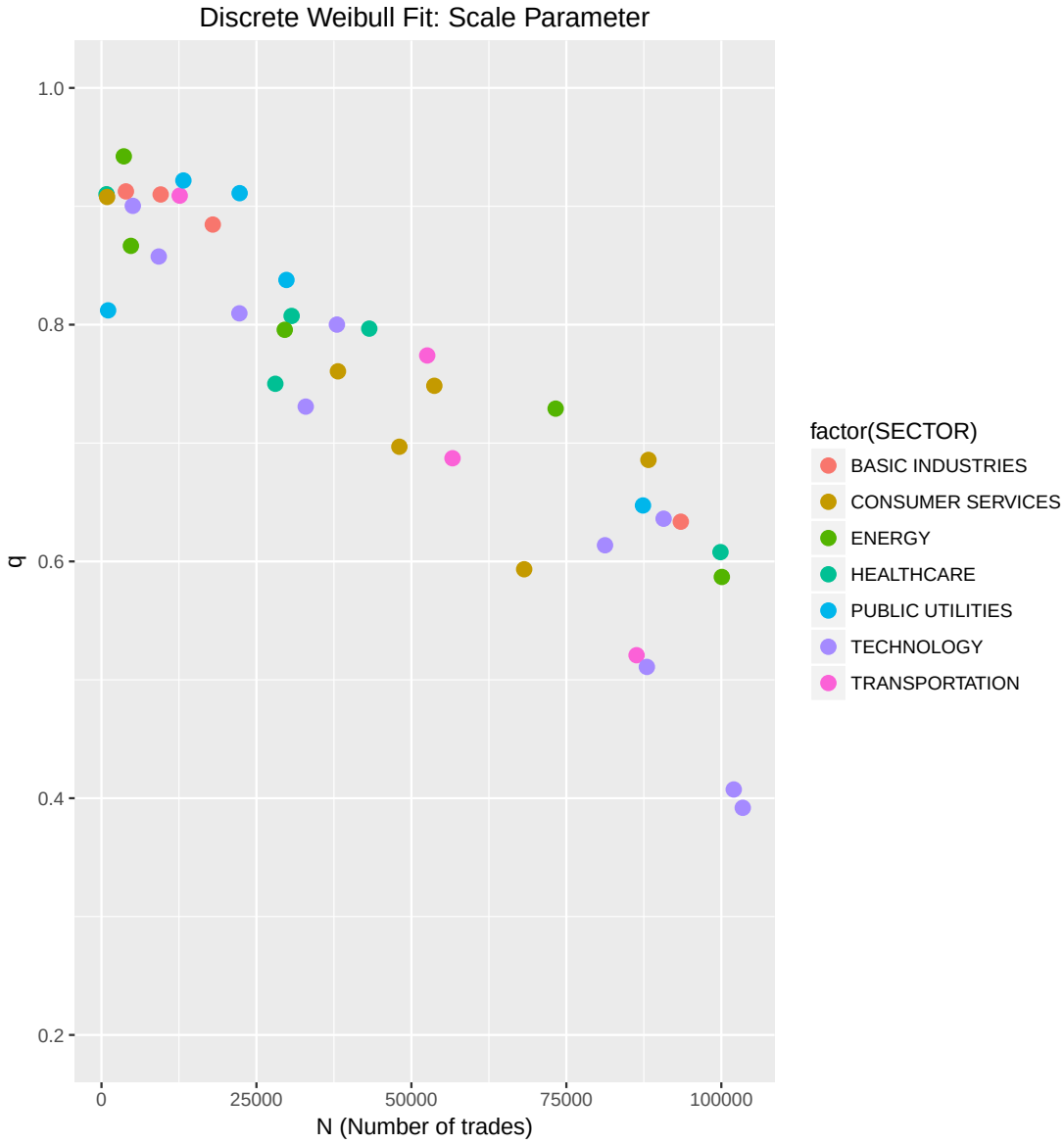

Figure 4.7: Discrete Weibull Fit Scale Parameters for securities from different sectors.

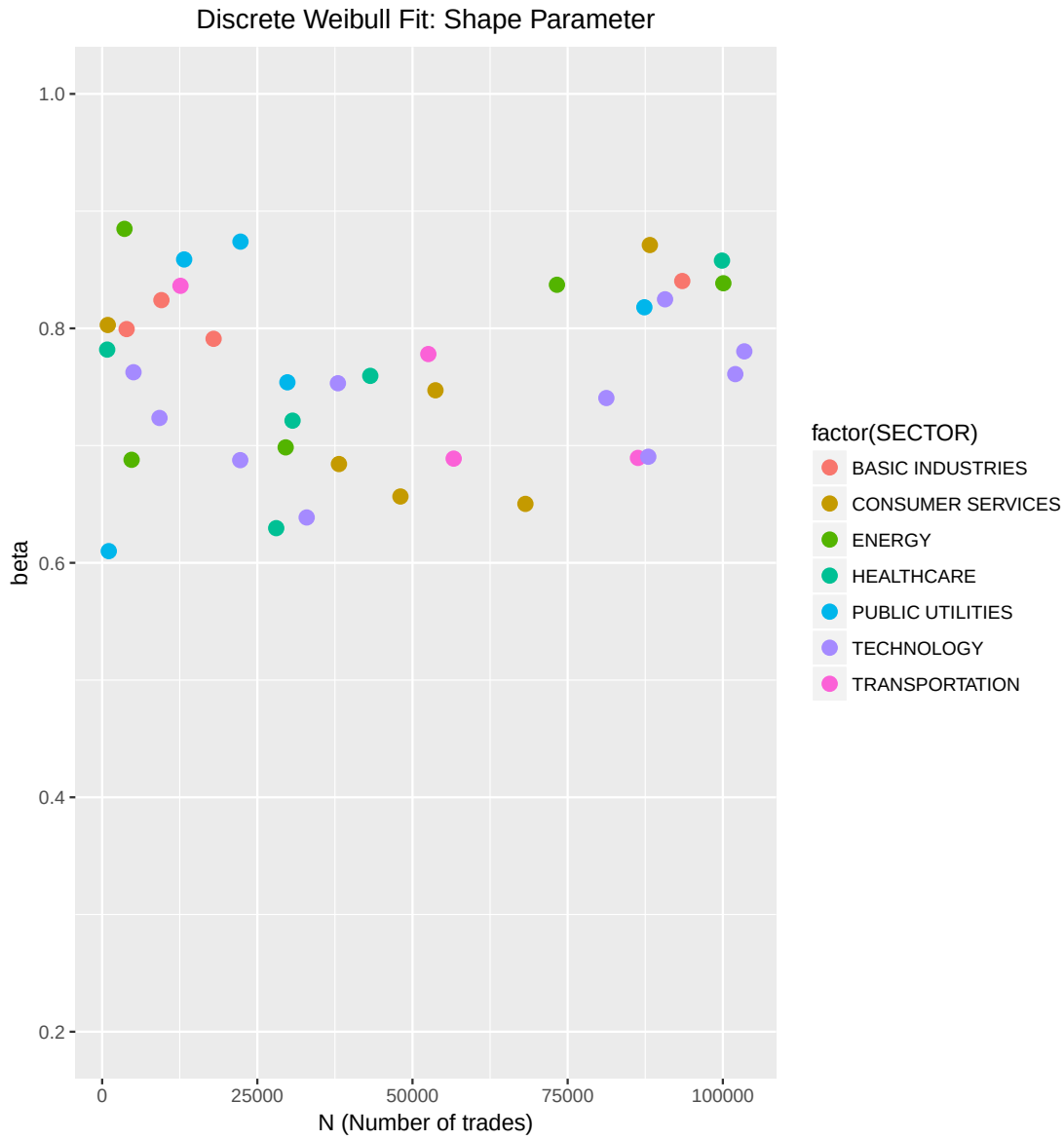

Figure 4.8: Discrete Weibull Fit Shape Parameters for securities from different sectors.

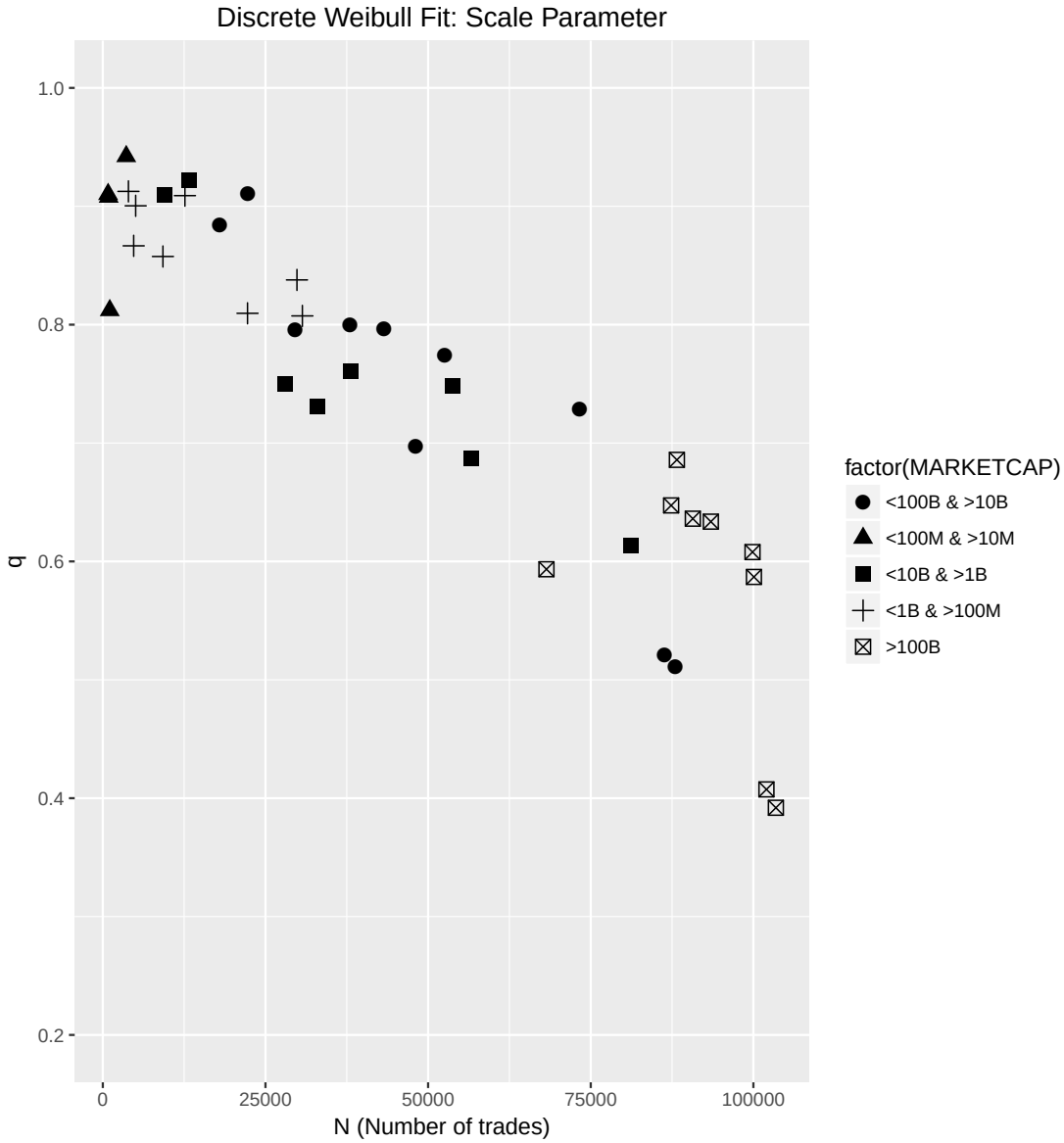

Figure 4.9: Discrete Weibull Fit Scale Parameters for securities from different market caps.

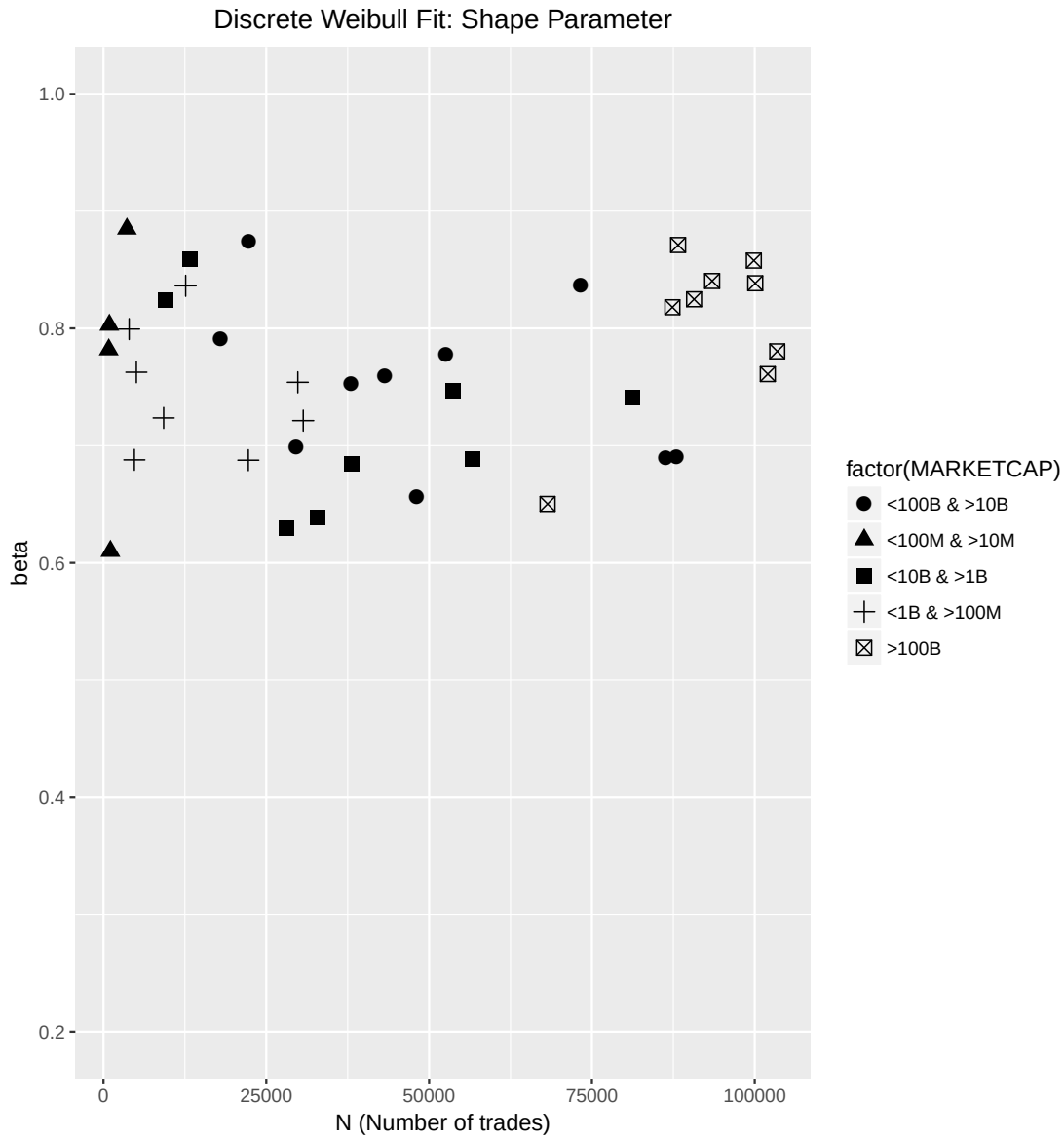

Figure 4.10: Discrete Weibull Fit Shape Parameters for securities from different market caps.

4.6 Conclusion

Combining Figure 4.7 with Figure 4.9 and Figure 4.8 with Figure 4.10 result in Figures 4.11 and 4.12 respectively. In this section, we analyzed the parameters of the the fits to to the Weibull distribution against market cap and industry. We do notice that the scale parameter, q, varies inversely with regards to the number of trades and inversely with respect to market cap. On the other hand, we don't really see any effect from market industry on the scale parameter. For the shape parameter, β, we don't see any dependence with regards to either number of trades, market industry or cap.

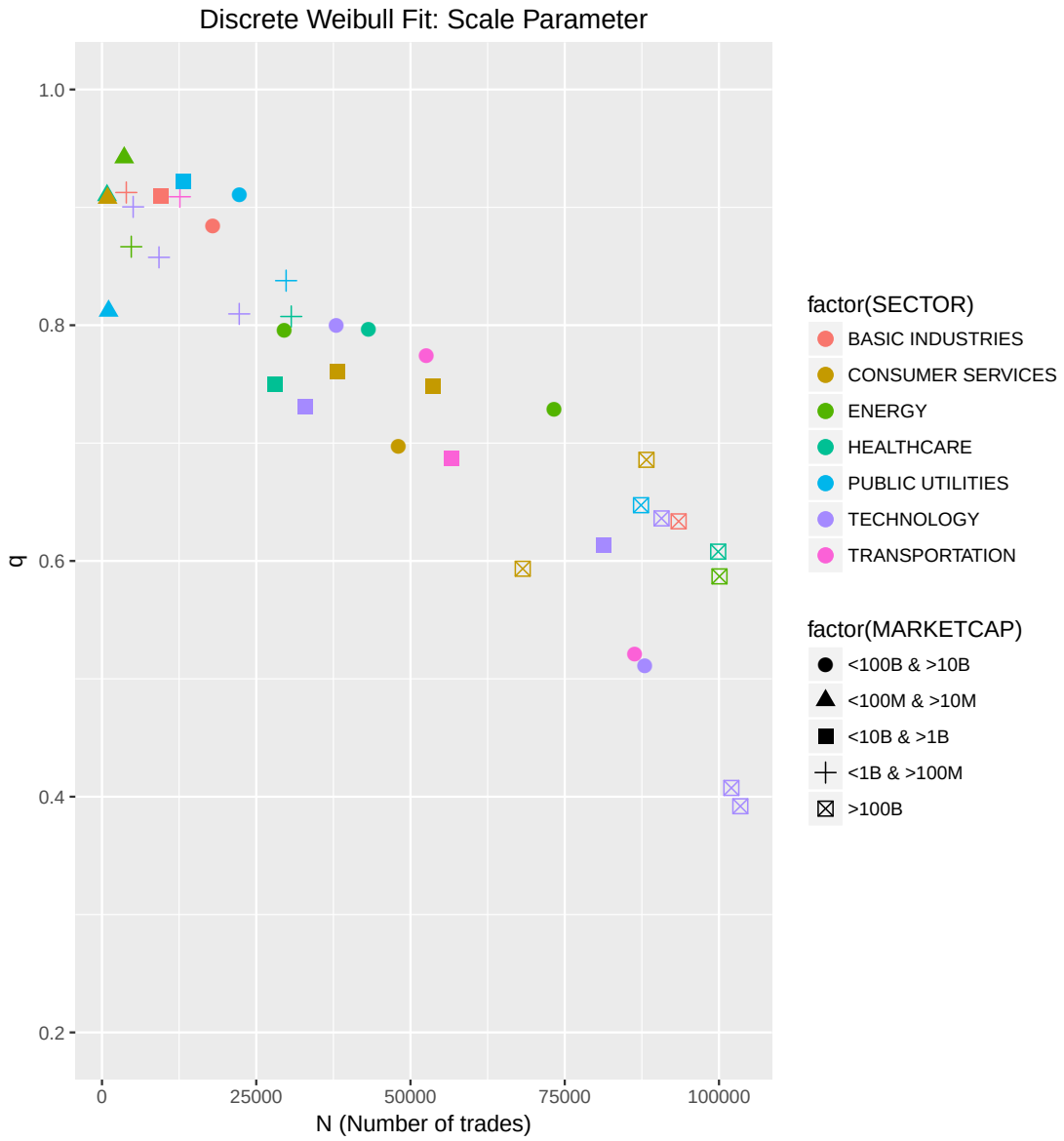

Figure 4.11: Discrete Weibull Fit Scale Parameters for securities from different sectors and different market caps.

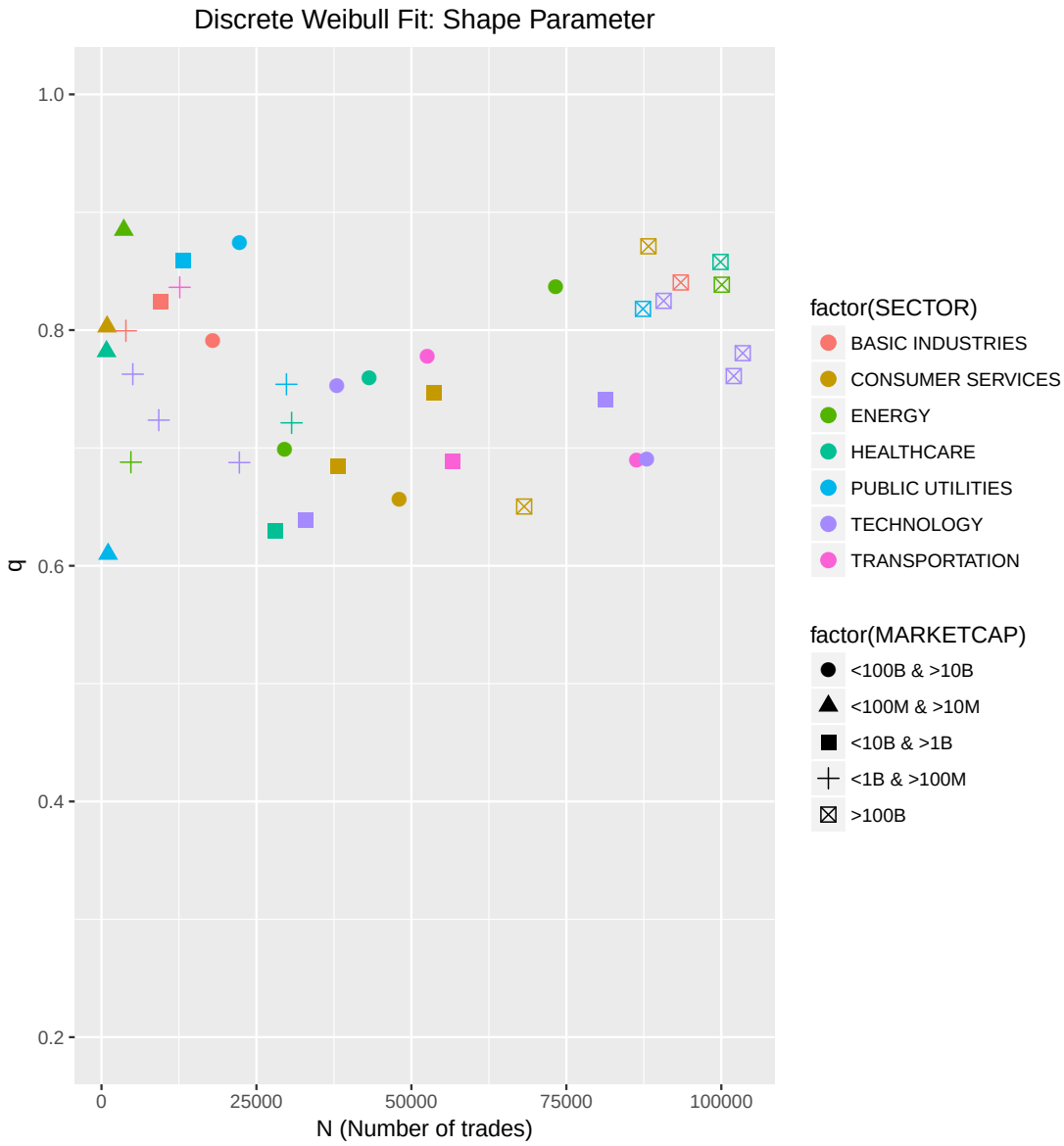

Figure 4.12: Discrete Weibull Fit Shape Parameters for securities from different sectors and different market caps.

CHAPTER 5

RESULTS II: INTERTRADE LIMIT ORDER BOOK ACTIVITY

Statistical properties of the Limit Order Books gathered a wide interest for both academic and practical purposes. At any time, shape of the order book convoluted with the order flow is the driving force behind what is observed as price impact [12, 13]. It was previously reported that statistics of incoming limit order prices distributed around the current bid/ask and the shape of the average order book seem to mimic the result of a zero-intelligence model [14]. Furthermore, stochastic models for dynamics of LOB start with the assumption of inflow of new orders being independent Poisson processes [23].

First, we will talk about the methods that will be used to accomplish the goals of this chapter. Then we will investigate the static Order Book Activity between two consecutive trades which would be useful in modeling the average shape of the Order Book.

Finally, we will investigate the dynamic Order Book Activity between two consecutive trades which would be useful in modeling Order Book Dynamics.

The ultimate purpose of this chapter is to shed light on market reaction in short and long term time scales to a particular trade at particular size. We will be achieving this goal by quantifying and statistically analyzing the distribution of order inflow at BID and ASK sides of the book as a function of relative price from the last trade price and time since last trade time.

5.1 Methods

In order to achieve what we proposed at the beginning of the Chapter 5, we will be analyzing TAQ data from one particular security, AMZN, from all exchanges. Moreover, we will be narrowing down our interest to the LOB Activity after a particular trade size $\Delta = 100$ and before the next trade happening at $\Delta t = 10s$. There will be $\Delta t = 10s$ available to investigate market reaction hence we will partition this time interval into 10 equal time intervals of each $1s$ and observe the LOB Activity in these time intervals and specified price intervals.

The price of a particular bid or ask quote is redefined as the difference from the last trade price being relative price. Furthermore, the relative prices are binned up to 10 ticks to reduce the noise. In other words, $\delta P \in [0, 0.05) \rightarrow \delta P = 0$, $\delta P \in [0.05, 0.15) \rightarrow \delta P = 0.1$ and so on the ASK side and $\delta P \in (-0.05, 0] \rightarrow \delta P = 0$, $\delta P \in (-0.15, 0.05] \rightarrow \delta P = -0.1$ on the BID side.

Finally, the data is conditioned on the previous trade sign being BUY or SELL event using Lee-Ready rule/algorithm [35].

5.2 Static Order Book Activity

LOB activity aggregated over time $t = 1, 2, \ldots, 9s$ for both limit order counts and limit order sizes are shown in Figures 5.1 and 5.2 respectively. The y-axes represent the relative BID prices in blue histograms and relative ASK prices in red histograms. Our first observation is the BID/ASK asymmetry after BUY/SELL trade in the distribution of relative prices.

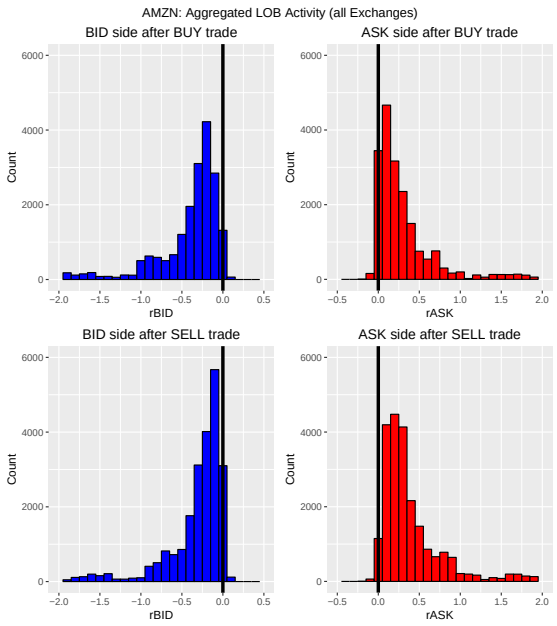

Figure 5.1: Aggregated LOB Activity (Order Count)

Previously, it was reported that incoming orders arrive more frequently within couple price tick distance away from best bid or ask [14]. Not only we observed a similar behavior in LOB activity but also we did notice BUY/SELL asymmetry in both BID and ASK sides of the order books for both order counts and order sizes.

Next, instead of aggregating over time, we only considered the early response which is $t = 1s$ after the trade in Figures 5.3 and 5.4.

Finally, we just considered the late response which is $t = 9s$ after the trade or $t = 1s$ before the next trade in Figures 5.5 and 5.6.

Hence, we observed BUY/SELL asymmetry in both BID and ASK sides of the order books at both early and late times for both order counts and order sizes.

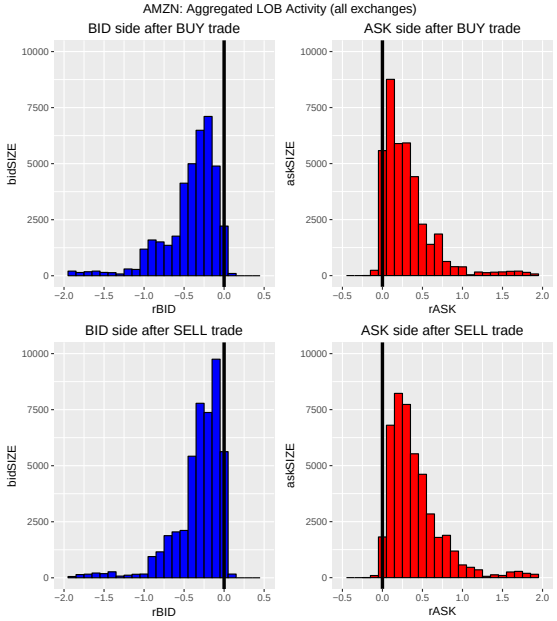

Figure 5.2: Aggregated LOB Activity (Order Size)

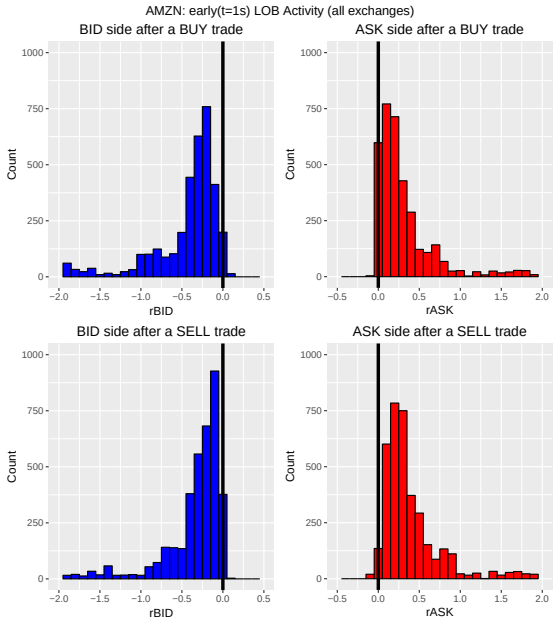

Figure 5.3: Early Time LOB Activity (Order Count)

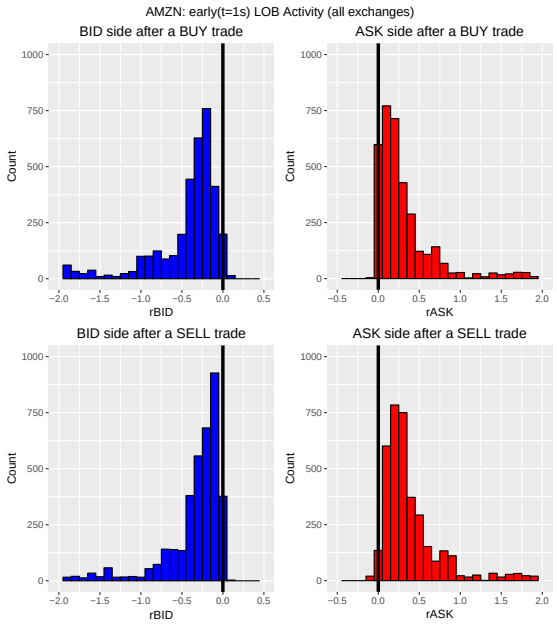

Figure 5.4: Early Time LOB Activity (Size)

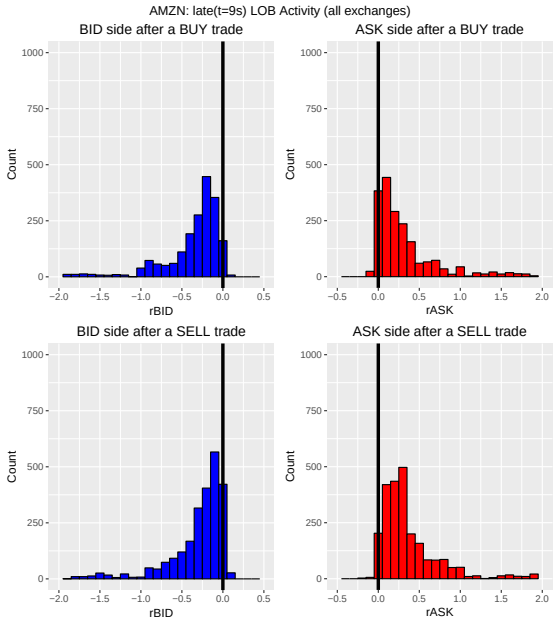

Figure 5.5: Late Time LOB Activity (Order Count)

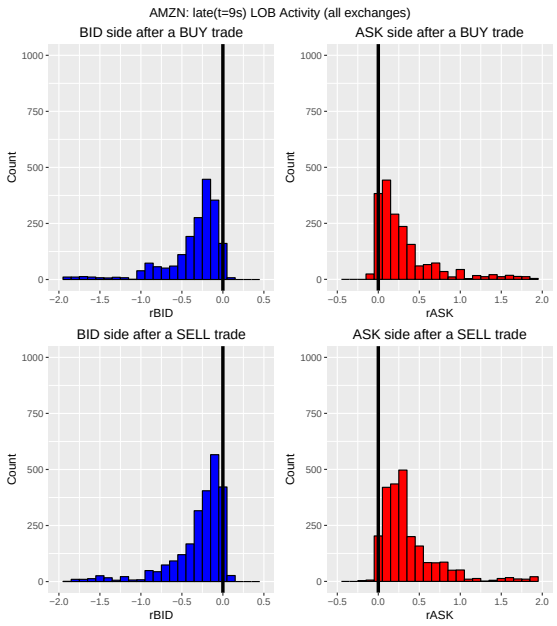

Figure 5.6: Late Time LOB Activity (Size)

5.3 Dynamic Order Book Activity

Modeling dynamics of LOB requires a proper description of incoming order flow. Most models in the literature assume incoming flow of orders are independent Poisson Processes [18, 19, 23]. This section investigates the adequacy of this assumption given the Order Book Activity.

Figure 5.7 shows LOB Activity at relative prices $\delta P \in (-0.05, 0]$ and $\delta P \in [0, 0.05)$ for BID/ASK side of the order books, respectively (specifically, after $t = 1s$ of a BUY or SELL trade while keeping the time window $\Delta t = 1s$). Row 1 of Figure 5.7 shows LOB Activity after a BUY trade, while Row 2 shows LOB Activity after a SELL trade. Column 1 of Figure 5.7 represents the BID side, and Column 2 the ASK side of LOB Activity.

Figure 5.8 shows LOB Activity at the same relative price window as Figure 5.7, but changing the time window to $t = 9s$ after the last trade.

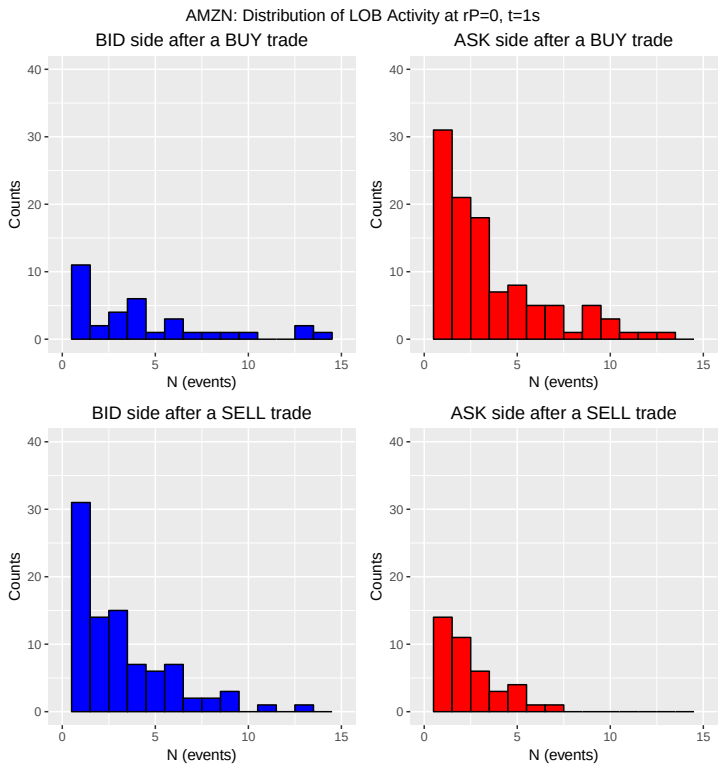

Figure 5.7: LOB Activity (N) at $\delta P \in (-0.05, 0]$ for the BID side and $\delta P \in [0, 0.05)$ for the ASK side after $t = 1s$ of BUY (top figures) and SELL (bottom figures) trades.

Figures 5.7 and 5.8 are similar to Figures 5.9 and 5.10; in the former, an event is defined to be the size of a limit order on the BID/ASK side, whereas an event in the latter is defined as the limit order arrival count regardless of the size of the incoming order.

5.4 Distributions for LOB Activity

Let X represent individual BID/ASK orders placed in a fixed interval of time t, assumed in literature to be Poisson distributed. Then,

$$P(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}, \; k = 0, 1, 2, \ldots \tag{5.1}$$

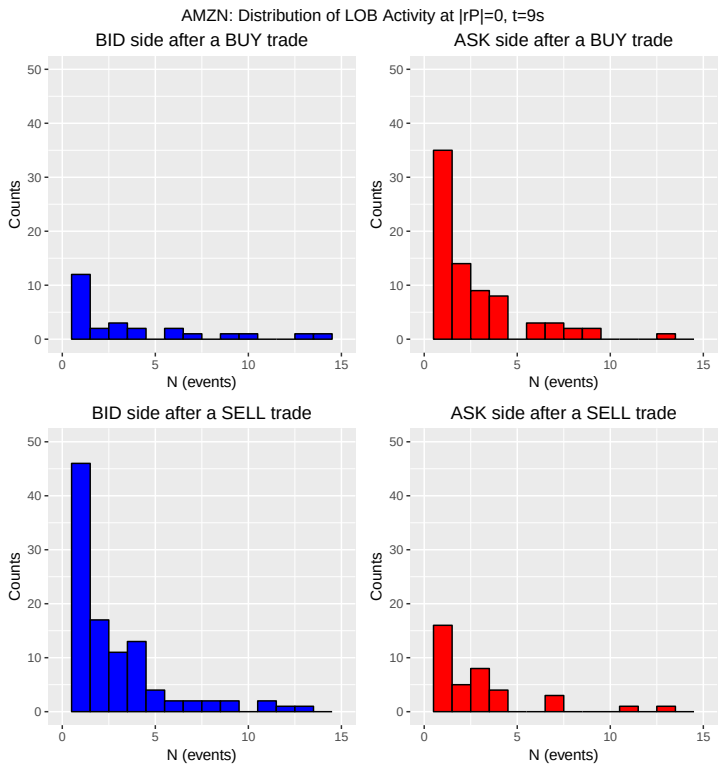

Figure 5.8: LOB Activity (N) at $\delta P \in (-0.05, 0]$ for the BID side and $\delta P \in [0, 0.05)$ for the ASK side after $t = 9s$ of BUY (top figures) and SELL (bottom figures) trades.

where parameter $\lambda > 0$ represents the average number of events that occur in time t and further, $EX = \text{Var}(X) = \lambda$. Moreover, it can be shown that sum of independent Poisson random variables is also Poisson distributed, namely

$$\text{if} \quad X_i \sim \text{Poisson}(\lambda_i), \quad \text{then} \quad \sum_{i=1}^{N} X_i \sim \text{Poisson}\Big(\sum_{i=1}^{N} \lambda_i\Big). \tag{5.2}$$

Table 5.1 summaries mean and variance statistics for LOB Activity for AMZN across all exchanges. LOB Activity at respective δP for BID/ASK sides. At all time intervals t on the BUY side, there appear to be an average of $2.5 \sim 5.5$ orders with a variance $4.6 \sim 31.6$. The variance of SELL side orders appear similarly overdispersed relative to Poisson distributed count events.

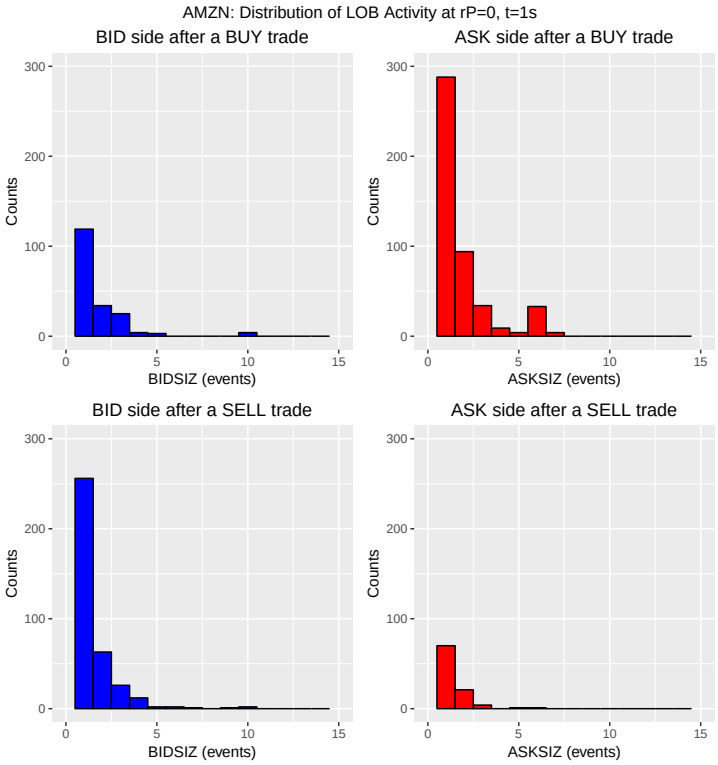

Figure 5.9: LOB Activity (SIZE) at $\delta P \in (-0.05, 0]$ for the BID side and $\delta P \in [0, 0.05)$ for the ASK side after $t = 1s$ of BUY (top figures) and SELL (bottom figures) trades.

5.4.1 Goodness of fit testing

To check if LOB activity are in fact Poisson distributed, we use a Pearson's χ^2 test on data from the price window $\delta P \in (-0.05, 0]$ on the BID side and $\delta P \in [0, 0.05)$ on the ASK side of the order book. This is applied to each time interval $t = 1, 2, \ldots, 9s$ after the last trade. The value of the test statistic is given by:

$$X^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}, \tag{5.3}$$

where O_i are the observed observed values of the random variable and E_i are the expected counts under the assumed (Poisson) distribution.

Asymptotically, $X^2 \xrightarrow{D} \chi^2(df)$, with $df = k - p - 1$; here $p = 1$ represents the signal parameter estimated in fitting a Poisson distribution to our data. Results

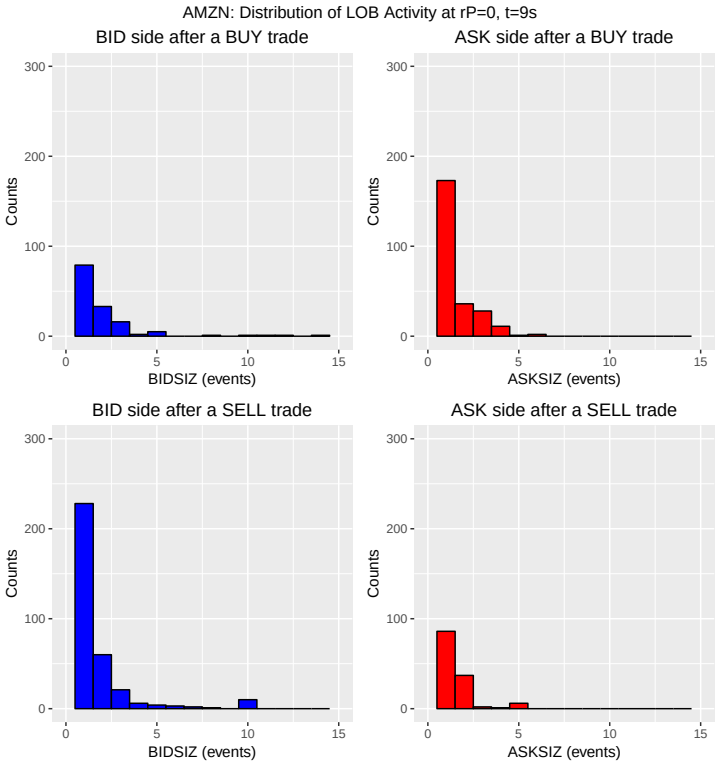

Figure 5.10: LOB Activity (SIZE) at $\delta P \in (-0.05, 0]$ for the BID side and $\delta P \in [0, 0.05)$ for the ASK side after $t = 9s$ of BUY (top figures) and SELL (bottom figures) trades.

are summarized in Table 5.2.

In summary, the majority of LOB model activity partitions, $t = 1, 2, \ldots, 9s$, have p-value ≈ 0 and generally much smaller than $\alpha = 0.05$, especially when we have sufficient degrees of freedom. For the aggregated LOB activity in time, we can confidently reject the null hypothesis for a majority of the LOB activities studied. Instead, one may consider data model which accomodates the observed overdispersion, such as the Negative-Binominal distribution NB$(r, \lambda/(\lambda + r))$:

$$P(X = k; \lambda, r) = \frac{\Gamma(r + k)}{\Gamma(r)(r + \lambda)^k} \frac{\lambda^k}{k!} \left(1 + \frac{\lambda}{r}\right)^{-r}, \tag{5.4}$$

with $EX = \lambda$ but $\text{Var}(X) = \lambda(1 + \lambda/r)$; here r controls the deviation from the Poisson model. More formally, note $\text{NB}(r, \lambda/(\lambda + r)) \overset{(r\to\infty)}{\longrightarrow} \text{Po}(\lambda)$.

		BID Side		ASK Side	
Time	Trade Sign	Mean	Variance	Mean	Variance
1s	BUY	5.25	29.16	4.24	28.24
1s	SELL	3.92	19.01	2.48	2.56
2s	BUY	4.02	31.57	2.66	6.01
2s	SELL	4.01	17.94	2.89	9.34
3s	BUY	2.56	4.56	2.95	7.95
3s	SELL	3.07	9.75	2.00	2.48
4s	BUY	3.46	12.34	2.77	5.53
4s	SELL	2.90	10.93	2.82	13.28
5s	BUY	2.35	5.96	2.96	6.74
5s	SELL	2.70	6.99	2.32	2.64
6s	BUY	3.79	29.80	2.95	5.66
6s	SELL	3.77	11.01	2.87	6.21
7s	BUY	4.28	20.71	3.11	9.52
7s	SELL	2.60	6.46	3.44	27.34
8s	BUY	4.65	15.17	3.20	7.63
8s	SELL	3.46	7.97	3.47	12.19
9s	BUY	5.00	35.56	3.18	15.15
9s	SELL	3.21	15.19	3.38	15.87

Table 5.1: Mean and Variance of LOB Activity (N) at $\delta P \in (-0.05, 0]$ for the BID side and $\delta P \in [0, 0.05)$ for the ASK side for AMZN.

5.5 Conclusion

In this chapter, we introduced LOB activity in between consecutive trades. We first demonstrated that static LOB activity reveals an interesting global asymmetry pattern in the bid-ask LOB state which depends on whether the previous event was a SELL or BUY event. Finally, we show that dynamic LOB activity between consecutive trades happening at a particular time and price interval ($\Delta t = 1s$, $\delta P = 0$) has a higher variance compared to the mean. Indeed, modeling this activity as Poisson yields a poor fit for the almost all of the LOB model partitions studied, as demonstrated by our goodness of fit tests.

		BID Side			ASK Side		
Time	Trade Sign	X^2	df	$p-val$	X^2	df	$p-val$
1s	BUY	8354.623	25	0	1088539	38	0
1s	SELL	13703.13	25	0	10.02133	6	0.1237568
2s	BUY	53787.76	29	0	277.667	12	0
2s	SELL	2049.593	19	0	247.9551	13	0
3s	BUY	33.21539	6	9.53e-06	406.6433	14	0
3s	SELL	1573.647	18	0	27.76305	7	0.0002427239
4s	BUY	228.2861	13	0	138.3279	10	0
4s	SELL	1558.505	17	0	720.6898	14	0
5s	BUY	137.1159	9	0	147.427	11	0
5s	SELL	1996.056	19	0	11.54619	6	0.07289312
6s	BUY	15980.11	25	0	144.6825	12	0
6s	SELL	330.1245	14	0	35.99011	9	3.980711e-05
7s	BUY	552.978	17	0	500.8469	14	0
7s	SELL	692.0526	15	0	14018.79	26	0
8s	BUY	286.221	16	0	148.6574	12	0
8s	SELL	160.9595	11	0	565.4715	17	0
9s	BUY	8453.585	24	0	18450.99	25	0
9s	SELL	80097.83	29	0	2294.271	20	0

Table 5.2: Pearson's χ^2 test results for LOB Activity (N) at $\delta P \in (-0.05, 0]$ for the BID side and $\delta P \in [0, 0.05)$ for the ASK side for AMZN.

CHAPTER 6

CONCLUSION AND FUTURE DIRECTION

The major contributions of this dissertation are as follows:

1. The waiting time between consecutive trades, intertrade time duration, follows a discrete Weibull distribution.
2. Shape and Scale parameters of Weibull distribution are independent of the security's market industry.
3. Most orders have a trade size of 100, independent of sector, especially for stocks with larger market capitalization.
4. There happens to be a BUY/SELL asymmetry in both BID and ASK sides of the Order Book.
5. Limit Order Book Activity is not following a Poisson Distribution unlike the assumption in most of the models in the literature.

6.1 Machine Learning and Future Direction

These findings improve our understanding about the dynamics behind trading and pave the way for novel research questions:

1. Can machine learning models for predicting various aspects of trading benefit from the inclusion of intertrade time durations and market capitalization as variables in the model?

Various machine learning techniques can be leveraged to explore these questions. Many machine models like logistic regression [1], k-nearest neighbors [10], multi-class support vector machines [26], reinforcement learning [39], and Bayesian networks [53] have been used for predictive tasks such as trade sign prediction. It is possible that these models can benefit from the inclusion of intratrade times and market capitalization. To explore this further, the first step would be to perform a correlation between intratrade times and trade sign as well as market capitalization and trade sign to see if the variables provide any predictive value. Subsequently, linear and logistic regression models could be built using with and without these variables to compare accuracy. Additionally, factor analysis could be used to extract a set of salient variables that can predict the trade sign and determine the weights of intratrade times and market capitalization toward prediction to see if the two variables provide novel predictive information or if they merely capture information that is already provided by other variables considered in the models. We discuss in more detail some of the possible machine learning models that have been used for predictive tasks and how our proposed variables could be included:

6.1.1 Logistic Regression

When embarking on any machine learning prediction model, a staple model to try is logistic regression. In [1], the authors used logistic regression to compute the probability of trading at different ask prices based on the time of day and found that the time of day is related to trade price. We propose to investigate the use of intraday timings and market capitalizations in a logistic regression model for predicting stock price or price direction, for example.

6.1.2 k-nearest neighbors

For classification problems, a simple yet effective classifier is the k-nearest neighbor. Given a training set of labeled class examples and a testing set of examples of unknown classes, the goal is to classify each example in the test set based on its similarity level to the examples in the training set. To this end, each training and testing example is initially represented by a set of features, and a similarity function is created that can quantify the similarity level between any two examples. To classify a test example, the similarity function is used to compute the similarity level between the test example and each training example; the k training examples closest to the test example are then identified and the most frequent class label of the k examples is returned as the predicted label for the test example. In [10], the authors predicted trade sign using kNN and found that it provided improved accuracy over logistic regression. The proposed model uses an inference model that selects a set of variables before performing the kNN. By simply including the intraday time and market capitalization as variables in the model, it would be possible to evaluate the effect of including such variables for prediction.

6.1.3 Stochastic modeling for limit order books

Stochastic approaches to model LOB dynamics have been proposed which can be used for short term price prediction and for designing optimal automated trading strategies [23]. A popular approach was proposed in [23] which models the limit order book as a stochastic process with Markovian dynamics. The model assumes that order and cancellation events are independent poisson pro-

cesses and intratrade times follow an exponential distribution [23]. Based on the findings of our work, market orders do not follow a poisson distribution and intratrade times were found to follow a discrete Weibull distribution. Other stochastic models have also been proposed which assume other distributions but, to the best of our knowledge, discrete Weibull distributions have not been used to model intratrade time. Our findings can be incorporated directly into these stochastic models to augment results.

An additional area of future work would be to model the LOB using Hidden Markov Models or Dynamic Bayesian Networks. The discrete Weibull distribution could be incorporated directly into the models using appropriate conditioning based on what the state space of the models represent.

6.1.4 Support Vector Machines

Support Vector Machines (SVM) are a common classification model in machine learning. A vector of attributes is used to represent each sample and the SVM learns the optimal linear hyperplane that separates the two classes of the SVM. For data that is not separable linearly, the SVM can be augmented to learn a non-linear boundary using the kernel trick. Specifically, instead of applying the SVM to the input data directly, the data is first mapped to a higher dimensional space before applying the SVM. In this way, the learned decision hyperplane is linear in the higher dimensional space but is a non-linear decision boundary in the original space. The authors in [26] and [33] use SVM to track limit order book dynamics. It is possible to extend these models to include the proposed intratrade timings and the market capitalization.

Apart from conventional machine learning techniques, an additional area that could be explored is time series classification methods.

6.1.5 Time Series Classification methods for trading

A plethora of methods exist for assessing the similarity between a pair of time series [5]. Most methods proposed in this area are tested on the UCR dataset repository [3] which consists of 85 datasets from various domains including healthcare, motion detection, and spectroscopy. However, there are no financial datasets in this repository. There is an opportunity to apply various time series classification methods to trade data for various prediction tasks. There are many categories of time series classification methods; of particular interest are interval-based methods that select features of the time series and use them classification [5]. Additionality, to explore the use of intra-trade timings, there are two time series classification techniques that already include timing values between timepoints for computing similarity, specifically, the Time Warp Edit Distance (TWED) and the Weighted Dynamic Time Warping (WDTW). These methods could be tested for various classification tasks pertaining to trade to test the efficacy of the methods. In addition to time series classification, some efforts have proposed temporal extensions of sequence alignment methods for healthcare where the goal is to include the timing between events when computing similarity [51, 50]]. These methods can also be extended to trade data with minimal changes.

1. Is a trade size of 100 an optimal choice or is there a better static or dynamically changing trading size? The commonly used trade size of 100 may

be a consequence of current automated trading algorithms. An obvious question is whether the trade size of 100 represents an optimal choice as it may be a consequence of automated trading algorithms. This question can be explored using machine learning models discussed in the previous section

2. Are there a separate set of dynamics and variables underlying trading at different market capitalizations? A straightforward way to explore this question is to create difference machine learning models for different capitalizations to see if it is possible to achieve higher accuracies by having different models. Moreover, as mentioned earlier, the market capitalization can be modelled directly within the machine learning algorithms, however, using a single value in the models may not be enough to represent its effects and it is possible that the factors that govern trading are quite different for smaller versus larger market capitalization stocks. In this case, it would be better to have independent models created for different market capitalizations.

APPENDIX A

SAS CODE

A.1 read_trade.SAS

Reading trades binary data, ct01u.bin and exporting as ct01u_AAL.csv

```
1  /* sas program to read TAQ trade binary files after year 2005.*/
2  options nocenter;
3  filename tbinfile '/folders/myfolders/2014/ct01u.bin' lrecl=19 recfm=f;
4
5  /* read the binary file */
6  data tmp1;
7    length TIME 8 PRICE 8 SIZE 8 G127 4 CORR 4 COND $2 EX $1.;
8    format time time.;
9    infile tbinfile firstobs=99678 obs=151063;
10     input @1 time IBR4. @5 price IBR4.4 @9 size IBR4. @13 g127 IBR2.  @15 corr IBR2.
           @17 cond $2. @19 ex $1.;
11 run;
12
13 /* export the binary file */
14 proc export data = tmp1
15   outfile='/folders/myfolders/2014/ct01u_AAL.csv' dbms=csv replace;
16 run;
```

read_trade.SAS

A.2 read_quote.SAS

Reading quotes binary data, cq01u.bin and exporting as cq01u_ACHN.csv

```
/* SAS program to read TAQ quote binary files starting 201108 - current.*/
options nocenter;
filename qbinfile '/folders/myfolders/2014/cq01u.bin'  lrecl=27 recfm=f;

data tmp1;
  length TIME 8 BID 8 OFR 8 BIDSIZ 8 OFRSIZ 8 MODE 4 EX $1 MMID $4;
  format time time.;
  infile qbinfile firstobs = 7510189  obs = 7566261;
  input @1 time IBR4. @5 bid IBR4.4 @9 ofr IBR4.4 @13 bidsiz IBR4.
      @17 ofrsiz IBR4. @21 mode IBR2. @ 23 ex $1. @24 mmid $4.;
run;

proc export data = tmp1  outfile='/folders/myfolders/2014/cq01u_ACHN.csv' dbms=csv
    replace; run;
```

read_quote.SAS

APPENDIX B

R CODE

B.1 index_extract.R

```
1  ###############################################################################
2  ################### Extract relevant rows from index file #####################
3  ###############################################################################
4  library(data.table)
5  datasource <- "~/Desktop/Research/indexFiles"
6  setwd(datasource)
7
8  #filename <- "ct01_all_idx.csv"
9  filename <- "cq01_all_idx.csv"
10 index_file <- fread(filename)
11 tickers <- c("AAL", "ACHN", "AET", "ALB", "ALDW", "AMD", "AMZN", "AWK", "BRS",
12             "CAR", "CEMI", "CLNE", "CLRO", "COP", "CVLT", "EMKR", "EXPE",
13             "FANG", "FDX", "GRPN", "HSTM", "IBM", "INTC", "JBLU", "JNJ", "LUB",
14             "MSFT", "MTX", "OMN", "PG", "POR", "ROYT", "SGEN", "T", "TWTR",
15             "VECO", "WMT", "XOM", "YELP")
16
17 index_file <- index_file[symbol%in% tickers, ][ ,let := unique(letters)[.GRP],
18                                                 by=date]
19 index_file <- index_file[ , .(let, symbol, begrec, endrec)]
20
21 #write.csv(index_file, paste(datasource, "/ct01_all_idx_sub.csv", sep=""))
22 write.csv(index_file, paste(datasource, "/cq01_all_idx_sub.csv", sep=""))
```

index_extract.R

B.2 fromCSVtoXTS.R

```
1  ###############################################################################
2  ############################## from CSV to XTS #################################
3  ###############################################################################
4  # CHECK: highfrequency_0.5.2 xts_0.9-5, zoo_1.7-10
5  library(highfrequency)
6  library(data.table)
7  setwd("~/Desktop/Research/raw_data")
8  stocks <- c("AAL", "ACHN", "AET", "ALB", "ALDW", "AMD", "AMZN", "AWK", "BRS",
9              "CAR", "CEMI", "CLNE", "CLRO", "COP", "CVLT", "EMKR", "EXPE",
10             "FANG", "FDX", "GRPN", "HSTM", "IBM", "INTC", "JBLU", "JNJ", "LUB",
11             "MSFT", "MTX", "OMN", "PG", "POR", "ROYT", "SGEN", "T", "TWTR",
12             "VECO", "WMT", "XOM", "YELP")
13 days <- c("a","b","c","d","e","f","g","h","i","j","k",
14          "l", "m","n","o", "p","q","r","s","t","u")
15 trade_days <- c("02", "03", "06", "07", "08", "09", "10",
16                "13", "14", "15", "16", "17","21", "22",
17                "23", "24", "27", "28", "29", "30", "31")
18 dates <- paste("201401", trade_days, sep="")
19
20 ###############################################################################
21 ############ add DATE to the TIME column for all CSV files  ###################
22 ###############################################################################
23 add_path <- function(d, flag) {
24   if (flag == "t") {
25     return(paste("_tradeCSV_v3/ct01", d, "_", ticker, ".csv", sep=""))
26   } else if (flag == "q") {
27     return(paste("_quoteCSV_v3/cq01", d, "_", ticker, ".csv", sep=""))
28   } else {
29     return(NULL)
30   }
31 }
32 add_date <- function(path_x, date_x) {
```

```
  old_csv <- data.table(read.csv(path_x))
  new_csv <- data.table(date_x[1], old_csv)
  setnames(new_csv, colnames(new_csv), c("DATE", colnames(old_csv)))
  write.csv(new_csv, path_x, row.names=FALSE)
}

path_trade <- sapply(as.list(days), add_path, flag="t")
path_quote <- sapply(as.list(days), add_path, flag="q")
mapply(add_date, path_trade, dates)
mapply(add_date, path_quote, dates)

################################################################################
#################### for each ticker: merge all CSV files ######################
################################################################################
merge_csv_all <- function(x, flag){
  data <- read.csv(x[1])
  for (i in 2:length(x))
    data <- rbind(data, read.csv(x[i]))
  if (flag == "t"){
    write.csv(data, paste(ticker, "_trades.csv", sep=""), row.names=FALSE)
  } else if (flag == "q") {
    write.csv(data, paste(ticker, "_quotes.csv", sep=""), row.names=FALSE)
  }
}
merge_csv_all(path_trade, "t")
merge_csv_all(path_quote, "q")

################################################################################
########################### Conversion: CSV -> XTS #############################
################################################################################
from <- "2014-01-02"
to <- "2014-01-31"
datasource <- "~/Desktop/Research/raw_data/_2014/_01"
```

```
datadestination <- "~/Desktop/Research/xts_data"

convert_all_trd <- function(x) {
  convert( from=from, to=to, datasource=datasource,
           datadestination=datadestination, trades = T,  quotes = F,
           ticker = x,
           dir = TRUE, extension = "csv",
           header = TRUE, tradecolnames=NULL,
           quotecolnames = NULL,
           format="%Y%m%d %H:%M:%S", onefile = TRUE )
}

convert_all_qte <- function(x) {
  convert( from=from, to=to, datasource=datasource,
           datadestination=datadestination, trades = F,  quotes = T,
           ticker = x,
           dir = TRUE, extension = "csv",
           header = TRUE, tradecolnames=NULL,
           quotecolnames = NULL,
           format="%Y%m%d %H:%M:%S", onefile = TRUE )
}

lapply(stocks, convert_all_trd)
lapply(stocks, convert_all_qte)
```

fromCSVtoXTS.R

B.3 highfrequency.R

```
################################################################################
###############   High Frequency Package to analyze TAQ data  #################
############### http://highfrequency.herokuapp.com/index.html #################
```

```
################################################################################
library(highfrequency)
library(timeDate)
library(data.table)

datadestination <- "~/Desktop/Research/xts_data"
setwd(datadestination)

################################################################################
############################### Load the data ##################################
################################################################################
stocks <- c("AAL", "ACHN", "AET", "ALB", "ALDW", "AMD", "AMZN", "AWK", "BRS",
            "CAR", "CEMI", "CLNE", "CLRO", "COP", "CVLT", "EMKR", "EXPE",
            "FANG", "FDX", "GRPN", "HSTM", "IBM", "INTC", "JBLU", "JNJ", "LUB",
            "MSFT", "MTX", "OMN", "PG", "POR", "ROYT", "SGEN", "T", "TWTR",
            "VECO", "WMT", "XOM", "YELP")
ticker <- stocks[1]
xts_data <- TAQLoad( tickers=ticker, from="2014-01-02", to="2014-01-31",
                     trades=TRUE, quotes=TRUE, datasource=datadestination)
trd_data <- xts_data[[1]]
qte_data <- xts_data[[2]]

################################################################################
#################### 3.1 Cleaning of highfrequency data ########################
################################################################################
trd_data <- exchangeHoursOnly(trd_data,
                              daybegin = "09:30:00", dayend = "16:00:00")
qte_data <- exchangeHoursOnly(qte_data,
                              daybegin = "09:30:00", dayend = "16:00:00")
############## delete observations where price or bid/ask is zero ##############
trd_data <- noZeroPrices(trd_data)
qte_data <- noZeroQuotes(qte_data)
############### delete entries with abnormal sales cond. "COND" ###############
```

```
trd_data <- salesCondition(trd_data)
##################### merge entries with same time stamp #####################
trd_data_merged <- mergeTradesSameTimestamp(trd_data,
                                            selection = "weightedaverage")
qte_data_merged <- mergeQuotesSameTimestamp(qte_data,
                                            selection = "weightedaverage")
######################### merge trades and quotes data ########################
taq_data <- matchTradesQuotes(trd_data_merged, qte_data_merged)
################ inferring trade direction, 1 (BUY), -1 (SELL) ################
trd_dir <- getTradeDirection(taq_data)
################# number of consecutive same trade directions #################
############ rle: computes length, values of runs of equal values #############
trd_dir_rep <- rle(paste(trd_dir, sep="|"))$lengths
trd_dir_val <- rle(paste(trd_dir, sep="|"))$values

################################################################################
############################## Function Definitions ############################
################################################################################

############ extract trade/quote times in seconds with the offset #############
extract_time <- function(x) {
  h_0 <- 9
  m_0 <- 30
  s_0 <- 0
  dh <- as.numeric(substr(x, 12, 13)) - h_0
  dm <- as.numeric(substr(x, 15, 16)) - m_0
  ds <- as.numeric(substr(x, 18, 19)) - s_0
  return( dh*60*60 + dm*60 + ds )
}

######################### extract the trade/quote DAY ##########################
extract_day <- function(x) {
  dy <- as.numeric(substr(x, 9, 10))
```

```
  return(dy)
}

################################################################################
########################## XTS/ZOO --> DATA TABLES ##############################
################################################################################

trd_data_dt <- as.data.table(trd_data)[ , .(extract_day(index),
                                            extract_time(index),
                                            as.character(levels(EX))[EX],
                                            as.numeric(levels(PRICE))[PRICE],
                                            as.numeric(levels(SIZE))[SIZE] )]
setnames(trd_data_dt, colnames(trd_data_dt),
         c("DAY", "TIME","EX", "PRICE", "SIZE"))

trd_time <- trd_data_dt[ , TIME]
trd_time_rep <- rle(paste(trd_time, sep="|"))$lengths
trd_time_val <- rle(paste(trd_time, sep="|"))$values
length(trd_dir) == length(trd_time_rep)
trd_dir_corr <- rep(trd_dir, trd_time_rep)

trd_data_dt <- trd_data_dt[ , DIR := trd_dir_corr]
qte_data_dt <- as.data.table(qte_data)[ , .(extract_day(index),
                                            extract_time(index),
                                            as.character(levels(EX))[EX],
                                            as.numeric(levels(BID))[BID],
                                            as.numeric(levels(BIDSIZ))[BIDSIZ],
                                            as.numeric(levels(OFR))[OFR],
                                            as.numeric(levels(OFRSIZ))[OFRSIZ])]
setnames(qte_data_dt, colnames(qte_data_dt),
         c("DAY", "TIME","EX", "BID", "BIDSIZ", "OFR", "OFRSIZ"))

################################################################################
```

```
########################## Function Definitions ##############################
##############################################################################

##### input:  trade data(data table), days(numeric vector)
##### output: datatable with particular day and trade time differences
time_diff <- function(x, N) {
  res <- data.table( x[ is.element(DAY, N), DAY[-1] ],
                     x[ is.element(DAY, N), EX[-1] ],
                     x[ is.element(DAY, N), TIME[-1] ],
                     x[ is.element(DAY, N), diff(TIME)],
                     x[ is.element(DAY, N), PRICE[-1]],
                     x[ is.element(DAY, N), SIZE[-1]],
                     x[ is.element(DAY, N), DIR[-1]],
                     x[ is.element(DAY, N), diff(DIR)] )
  setnames(res, colnames(res),
           c("DAY","EX", "TIME", "DTIME", "PRICE", "SIZE", "DIR", "MOM"))
  return(res)
}

days <- c(2:3,6:10,13:17,21:24,27:31)
tau <- 0
time_int <- time_diff(trd_data_dt, days)[DTIME>=tau,
                                         list(DAY, EX, TIME, TIME-DTIME,
                                              DTIME, PRICE, SIZE, DIR, MOM)]
setnames(time_int, colnames(time_int),
         c("DAY","EX", "tTIME", "ptTIME", "DTIME",
           "PRICE", "SIZE", "DIR", "MOM"))
time_int <- time_int[ , SYMBOL := ticker]

path_destination <- "~/Desktop/Research/RData/"

save(time_int, file=paste(path_destination,
                          "time_int_", ticker, ".RData", sep=""))
```

```
save(trd_data_dt, file=paste(path_destination,
                             "trd_data_dt_", ticker, ".RData", sep=""))
save(qte_data_dt, file=paste(path_destination,
                             "qte_data_dt_", ticker, ".RData", sep=""))
```

highfreq.R

B.4 time_duration_fit.R

```
################################################################################
################### INTERTRADE TIME DURATION: model and fits ###################
################################################################################
rm(list=ls())
library(highfrequency)
library(timeDate)
library(ggplot2)
library(data.table)
library(plyr)
library(fitdistrplus)
library(FAdist)
library(gamlss)
library(DiscreteWeibull)
library(actuar)
library(degreenet)
theme_update(plot.title = element_text(hjust = 0.5))
datasource <- "~/Desktop/Research/RData"
setwd(datasource)

################################################################################
###################### Load data: various stocks at once #######################
################################################################################

```

```
stocks <- c("AAL", "ACHN", "AET", "ALB", "ALDW", "AMD", "AMZN", "AWK", "BRS",
            "CAR", "CEMI", "CLNE", "CLRO", "COP", "CVLT", "EMKR", "EXPE",
            "FANG", "FDX", "GRPN", "HSTM", "IBM", "INTC", "JBLU", "JNJ", "LUB",
            "MSFT", "MTX", "OMN", "PG", "POR", "ROYT", "SGEN", "T", "TWTR",
            "VECO", "WMT", "XOM", "YELP")

data_name <- function(x){
  return (paste("time_int_", x, ".RData", sep=""))
}
helper <- function(x){
  load(file=x)
  get(ls()[ls()!="filename"])
}
all_files <- unlist(lapply(stocks, data_name))
dt_list <- lapply(all_files, helper)
time_int <- rbindlist(dt_list)

################################################################################
############## CATEGORIZE securities for sector and market cap ################
################################################################################

tech_stock <- c("MSFT", "INTC", "IBM", "TWTR", "GRPN", "HSTM", "AMD", "CVLT",
                "EMKR", "VECO")
cons_stock <- c("AMZN", "WMT", "EXPE", "CAR", "YELP", "LUB")
transp_stock <- c("UNP", "FDX", "AAL", "JBLU", "BRS")
basic_stock <- c("PG", "ALB", "MTX", "OMN")
public_stock <- c("T", "AWK", "POR", "CLNE", "CLRO")
health_stock <- c("JNJ", "AET", "SGEN", "ACHN", "CEMI")
energy_stock <- c("XOM", "COP", "FANG", "ALDW", "ROYT")

cap_1 <- c("MSFT", "INTC", "IBM", "AMZN", "WMT", "PG", "T", "JNJ", "XOM") #>100B
cap_2 <- c("AMD", "TWTR", "EXPE", "FDX", "AAL", "ALB",
           "AWK", "AET", "COP", "FANG") #10B-100B
```

```
cap_3 <- c("CVLT", "GRPN", "CAR", "YELP", "JBLU", "MTX", "POR", "SGEN") #1B-10B
cap_4 <- c("VECO", "EMKR", "HSTM", "BRS", "OMN", "CLNE", "ACHN", "ALDW") #0.1B-1B
cap_5 <- c("LUB", "CLRO", "CEMI", "ROYT") #<0.1B

match_sector <- function(x){
  if (x%in%tech_stock) return("TECHNOLOGY")
  else if (x%in%cons_stock) return("CONSUMER SERVICES")
  else if (x%in%transp_stock) return("TRANSPORTATION")
  else if (x%in%basic_stock) return("BASIC INDUSTRIES")
  else if (x%in%public_stock) return("PUBLIC UTILITIES")
  else if (x%in%health_stock) return("HEALTHCARE")
  else if (x%in%energy_stock) return("ENERGY")
  else return("NA")
}

match_market_cap <- function(x){
  if (x%in%cap_1) return(">100B")
  else if (x%in%cap_2) return("<100B & >10B")
  else if (x%in%cap_3) return("<10B & >1B")
  else if (x%in%cap_4) return("<1B & >100M")
  else if (x%in%cap_5) return("<100M & >10M")
  else return("NA")
}

################################################################################
################################################################################
#################### HISTOGRAMS: INTERTRADE TIME DURATION ######################
################################################################################
################################################################################

ticker <- c("IBM")
a1 <- ggplot(time_int[DTIME>=1 & SYMBOL==ticker, ], aes(x=DTIME)) +
  geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5) +
```

```
  ggtitle(paste("Intertrade time duration: ", ticker, sep="")) +
  labs(x="Time (s)", y="Count") + xlim(0,60)
ggsave(filename=paste("hist_ITT_", ticker, ".pdf", sep=""), plot=a1)

a2 <- ggplot(time_int[DTIME>=1 & SIZE==100 & SYMBOL==ticker, ], aes(x=DTIME)) +
  geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5) +
  ggtitle(paste("Intertrade time duration (SIZE==100): ", ticker, sep="")) +
  labs(x="Time (s)", y="Count") + xlim(0,60)
ggsave(filename=paste("hist_ITT_sz100_", ticker, ".pdf", sep=""), plot=a2)

######################### HISTOGRAMS: ITT multiplots ##########################

tickers <- c("PG", "AMZN", "CAR", "HSTM")
b1 <- ggplot(time_int[DTIME>=1 & SYMBOL==tickers[1], ], aes(x=DTIME)) +
  geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
                 aes(y=..density..)) +
  ggtitle(paste("Intertrade time duration: ", tickers[1], sep="")) +
  labs(x="Time (s)", y="Density") + xlim(0,60)

b2 <- ggplot(time_int[DTIME>=1 & SYMBOL==tickers[2], ], aes(x=DTIME)) +
  geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
                 aes(y=..density..)) +
  ggtitle(paste("Intertrade time duration: ", tickers[2], sep="")) +
  labs(x="Time (s)", y="Density") + xlim(0,60)

b3 <- ggplot(time_int[DTIME>=1 & SYMBOL==tickers[3], ], aes(x=DTIME)) +
  geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
                 aes(y=..density..)) +
  ggtitle(paste("Intertrade time duration: ", tickers[3], sep="")) +
  labs(x="Time (s)", y="Density") + xlim(0,60)

b4 <- ggplot(time_int[DTIME>=1 & SYMBOL==tickers[4], ], aes(x=DTIME)) +
  geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
```

```
123                 aes(y=..density..)) +
124   ggtitle(paste("Intertrade time duration: ", tickers[4], sep="")) +
125   labs(x="Time (s)", y="Density") + xlim(0,60)
126
127 plt <- grid.arrange( b1, b2, b3, b4, ncol = 2, nrow=2,
128                      top = "Intertrade Time Duration ")
129
130 ggsave(filename=paste("hist_ITT_multi", ".pdf", sep=""), plot=plt)
131 #ggsave(filename=paste("hist_ITT_sz100_multi", ".pdf", sep=""), plot=plt)
132
133 ##############################################################################
134 ########################### Summary and Statistics ###########################
135 ##############################################################################
136
137 mean_size <- time_int[ DTIME>=1, round(mean(SIZE),2), by=SYMBOL]
138 setnames(mean_size, colnames(mean_size), c("SYMBOL", "mSIZE"))
139
140 ########################## MEAN trade time duration ##########################
141 mean_dtime <- time_int[ DTIME>=1 , round(mean(DTIME),2), by=SYMBOL]
142 #mean_dtime <- time_int[ , mean(DTIME), by=SYMBOL]
143 setnames(mean_dtime, colnames(mean_dtime), c("SYMBOL", "mDTIME"))
144
145 ########################## TOTAL VOLUME ##########################
146 total_vol <- time_int[ DTIME>=1, round(sum(SIZE)/1e6,3), by=SYMBOL]
147 setnames(total_vol, colnames(total_vol), c("SYMBOL", "V"))
148
149 ########################## TOTAL NUMBER OF TRADES ##########################
150 num_trades <- time_int[ DTIME>=1, .N, by=SYMBOL]
151 setnames(num_trades, colnames(num_trades), c("SYMBOL", "N"))
152
153 ########################## MIN/MAX TRADE PRICE ##########################
154 min_price <- time_int[ DTIME>=1, min(PRICE), by=SYMBOL]
155 setnames(min_price, colnames(min_price), c("SYMBOL", "minP"))
```

```
156 max_price <- time_int[ DTIME>=1, max(PRICE), by=SYMBOL]
157 setnames(max_price, colnames(max_price), c("SYMBOL", "maxP"))
158
159 ################################## SUMMARY ###################################
160 make_stats <- function(x) {
161   stat_x <- data.table(x, num_trades[SYMBOL==x, N], total_vol[SYMBOL==x, V ],
162                           round(mean_dtime[SYMBOL==x, mDTIME], 2),
163                           round(mean_size[SYMBOL==x, mSIZE], 2),
164                           round(min_price[SYMBOL==x, minP], 2),
165                           round(max_price[SYMBOL==x, maxP], 2)
166                       )
167   setnames(stat_x, colnames(stat_x),
168            c("SYMBOL", "N", "V", "mDTIME", "mSIZE", "minP", "maxP"))
169   return(stat_x)
170 }
171
172 stocks_stats <- rbindlist(lapply(stocks, make_stats))
173 write.csv(stocks_stats, file = "_stocks_stats.csv")
174
175 make_sz_per <- function(x){
176   sz_1 <- 10
177   sz_2 <- 100
178   sz_3 <- 1000
179   norm_0 <- time_int[DTIME < 1 & SYMBOL == x, .N]
180   total_0_1 <- time_int[DTIME< 1 & SIZE<sz_1 & SYMBOL == x, .N]
181   total_0_2 <- time_int[DTIME< 1 & SIZE>=sz_1 & SIZE<sz_2 & SYMBOL == x, .N]
182   total_0_3 <- time_int[DTIME< 1 & SIZE==sz_2 & SYMBOL == x, .N]
183   total_0_4 <- time_int[DTIME< 1 & SIZE>sz_2 & SIZE<=sz_3 & SYMBOL == x, .N]
184   norm_1 <- time_int[DTIME >= 1  & SYMBOL == x, .N]
185   total_1_1 <- time_int[DTIME >= 1 & SIZE<sz_1 & SYMBOL == x, .N]
186   total_1_2 <- time_int[DTIME >= 1 & SIZE>=sz_1 & SIZE<sz_2 & SYMBOL == x, .N]
187   total_1_3 <- time_int[DTIME >= 1 & SIZE==sz_2 & SYMBOL == x, .N]
188   total_1_4 <- time_int[DTIME >= 1 & SIZE>sz_2 & SIZE<=sz_3 & SYMBOL == x, .N]
```

```
  sz_per <- data.table(x, round(100*total_0_1/norm_0, 2),
                          round(100*total_0_2/norm_0, 2),
                          round(100*total_0_3/norm_0, 2),
                          round(100*total_0_4/norm_0, 2),
                          round(100*total_1_1/norm_1, 2),
                          round(100*total_1_2/norm_1, 2),
                          round(100*total_1_3/norm_1, 2),
                          round(100*total_1_4/norm_1, 2) )
  setnames(sz_per, colnames(sz_per),
           c("SYMBOL", "p1", "p2", "p3", "p4", "q1", "q2", "q3", "q4"))
  return(sz_per)
}

stocks_sz_per <- rbindlist(lapply(stocks, make_sz_per))
write.csv(stocks_sz_per, file="_stocks_sz_per.csv")

################################################################################
########################## ORDER SIZE Scatter Plot ############################
################################################################################

sz_per <- as.data.table(read.csv("_stocks_sz_per.csv"))
sz_per <- sz_per[ , .(p3, q3), by="SYMBOL"]
sz_per[ , SECTOR := as.character(lapply(unlist(.SD), match_sector)),
        .SDcols="SYMBOL"]
sz_per[ , MARKETCAP := as.character(lapply(unlist(.SD), match_market_cap)),
        .SDcols="SYMBOL"]
st_stats <- as.data.table(read.csv("_stocks_stats.csv"))

datum <- sz_per[ st_stats, on = .(SYMBOL),
                 ":=" ( N = N, V = V, mDTIME = mDTIME, mSIZE = mSIZE,
                        p3 = p3, q3 = q3, SECTOR = SECTOR,
                        MARKETCAP=MARKETCAP)]

```

```
e1 <- ggplot(data=datum, aes(x=log(N), y=p3)) +
  geom_point(aes(color = factor(SECTOR), shape = factor(MARKETCAP)), size = 3) +
  ggtitle(paste("Percent of trades at SIZE=100, FAST time scale ", sep="")) +
  labs(x="log (Number of trades)", y="Percent") + ylim(0,100)
ggsave(filename=paste("TR_per_logN_fast_mcap_sec", ".pdf", sep=""), plot=e1)

e2 <- ggplot(data=datum, aes(x=log(N), y=q3)) +
  geom_point(aes(color = factor(SECTOR), shape = factor(MARKETCAP)), size = 3) +
  ggtitle(paste("Percent of trades at SIZE=100, SLOW time scale ", sep="")) +
  labs(x="log (Number of trades)", y="Percent") + ylim(0,100)
ggsave(filename=paste("TR_per_logN_slow_mcap_sec", ".pdf", sep=""), plot=e2)

################################################################################
############################## DISCRETE DIST FITS ##############################
################################################################################
tau_int <- 1:60
data <-  time_int[ DTIME%in%tau_int , as.numeric(.N),
                   by=.(SYMBOL, DTIME)][order(SYMBOL, DTIME)]
setnames(data, colnames(data), c("SYMBOL", "DTIME", "N"))

################################################################################
############################## DATA and SIMULATION #############################
################################################################################

zardoz <- function(ticker) {
  ticker_data <- data[ SYMBOL==ticker, rep(DTIME,N)]
  wb_model <- fitdist(ticker_data, "dweibull", start=list(q=0.8, beta=1))
  gm_model <- fitdist(ticker_data, "ztgeom", start=list(prob=0.5))
  wb_sim <- rdweibull(length(ticker_data), wb_model$estimate[1],
                      wb_model$estimate[2])
  wb_sim <- as.data.table(table(wb_sim))
  setnames(wb_sim, c("DTIME", "N"))
  wb_sim <- wb_sim[ , DTIME := as.integer(DTIME)]
```

```
  N_tot_wb <- wb_sim[, sum(N)]
  wb_sim <- wb_sim[ , dN := (N/N_tot_wb)]
  wb_sim[ , SYMBOL := ticker]
  wb_sim[ , MODEL := "weibull"]

  gm_sim <- rztgeom(length(ticker_data), gm_model$estimate[1])
  gm_sim <- as.data.table(table(gm_sim))
  setnames(gm_sim, c("DTIME", "N"))
  gm_sim <- gm_sim[ , DTIME := as.integer(DTIME)]
  N_tot_gm <- gm_sim[, sum(N)]
  gm_sim <- gm_sim[ , dN := (N/N_tot_gm)]
  gm_sim[ , SYMBOL := ticker]
  gm_sim[ , MODEL := "geom"]

  res <- rbind(wb_sim, gm_sim)
  return(res)
}

tickers <- c("PG", "AMZN", "CAR", "HSTM")
#tickers <- c( "JNJ", "FDX", "FANG", "CLNE")
sim_model <- rbindlist(lapply(tickers, zardoz))


d1 <- ggplot(data=time_int[SYMBOL==tickers[1] & DTIME%in%tau_int, ],
             aes(x=DTIME)) +
  geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
                 aes(y=..density..)) +
  geom_point(data=sim_model[SYMBOL==tickers[1] & MODEL=="weibull", ],
             aes(x=DTIME, y=dN, color="weibull"), size=2.5) +
  geom_point(data=sim_model[SYMBOL==tickers[1] & MODEL=="geom", ],
             aes(x=DTIME, y=dN, color="geom"), size=2.5) +
  ggtitle(paste("Intertrade time duration: ", tickers[1], sep="")) +
  labs(x="Time (s)", y="Density") + xlim(0,60)
```

```
288
289 d2 <- ggplot(data=time_int[SYMBOL==tickers[2] & DTIME%in%tau_int, ],
290             aes(x=DTIME)) +
291   geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
292                  aes(y=..density..)) +
293   geom_point(data=sim_model[SYMBOL==tickers[2] & MODEL=="weibull", ]
294             , aes(x=DTIME, y=dN, color="weibull"), size=2.5) +
295   geom_point(data=sim_model[SYMBOL==tickers[2] & MODEL=="geom", ]
296             , aes(x=DTIME, y=dN, color="geom"), size=2.5) +
297   ggtitle(paste("Intertrade time duration: ", tickers[2], sep="")) +
298   labs(x="Time (s)", y="Density") + xlim(0,60)
299
300 d3 <- ggplot(data=time_int[SYMBOL==tickers[3] & DTIME%in%tau_int, ],
301             aes(x=DTIME)) +
302   geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
303                  aes(y=..density..)) +
304   geom_point(data=sim_model[SYMBOL==tickers[3] & MODEL=="weibull", ],
305             aes(x=DTIME, y=dN, color="weibull"), size=2.5) +
306   geom_point(data=sim_model[SYMBOL==tickers[3] & MODEL=="geom", ],
307             aes(x=DTIME, y=dN, color="geom"), size=2.5) +
308   ggtitle(paste("Intertrade time duration: ", tickers[3], sep="")) +
309   labs(x="Time (s)", y="Density") + xlim(0,60)
310
311 d4 <- ggplot(data=time_int[SYMBOL==tickers[4] & DTIME%in%tau_int, ],
312             aes(x=DTIME)) +
313   geom_histogram(binwidth=1, fill="blue", color="black", boundary=0.5,
314                  aes(y=..density..)) +
315   geom_point(data=sim_model[SYMBOL==tickers[4] & MODEL=="weibull", ],
316             aes(x=DTIME, y=dN, color="weibull"), size=2.5) +
317   geom_point(data=sim_model[SYMBOL==tickers[4] & MODEL=="geom", ],
318             aes(x=DTIME, y=dN, color="geom"), size=2.5) +
319   ggtitle(paste("Intertrade time duration: ", tickers[4], sep="")) +
320   labs(x="Time (s)", y="Density") + xlim(0,60)
```

```
321
322 plt <- grid.arrange( d1, d2, d3, d4, ncol = 2, nrow=2,
323                      top = "Intertrade Time Duration")
324
325 ggsave(filename=paste("hist_ITT_multi_fits_v1", ".pdf", sep=""), plot=plt)
326
327 ###############################################################################
328 ############################ PARAMETER EXTRACTION ##############################
329 ###############################################################################
330
331 discrete_fits <- function(ticker){
332   model1 <- fitdist(data[SYMBOL==ticker, rep(DTIME,N)], "dweibull",
333                     start=list(q=0.8, beta=1))
334   model2 <- fitdist(data[SYMBOL==ticker, rep(DTIME,N)], "ztgeom",
335                     start=list(prob=0.5))
336   D <- 2*(model1$loglik-model2$loglik)
337   res <- data.table(ticker, round(model1$estimate[1],4),
338                     round(model1$estimate[2],4), model1$loglik, model1$aic,
339                     round(model2$estimate[1],4), model2$loglik, model2$aic,
340                     pchisq(D, df=1, lower.tail=FALSE))
341   setnames(res, c("SYMBOL", "q", "beta", "w_loglik", "w_aic",
342                   "p", "g_loglik", "g_aic", "p-val"))
343   return(res)
344 }
345
346 conf_int <- function(ticker) {
347   ticker_data <- data[ SYMBOL==ticker, rep(DTIME,N)]
348   wb_model <- fitdist(ticker_data, "dweibull", start=list(q=0.8, beta=1))
349   var <- varFisher(ticker_data, zero=FALSE)
350   beta_ll <- round(wb_model$estimate[2] -
351                      1.96 * sqrt( var$InvFisherInfMatrix[2,2]
352                                   / length(ticker_data)), 4)
353   beta_ul <- round(wb_model$estimate[2] +
```

```
                    1.96 * sqrt( var$InvFisherInfMatrix[2,2]
                                 / length(ticker_data)), 4)
  q_ll <- round(wb_model$estimate[1] -
                    1.96 * sqrt( var$InvFisherInfMatrix[1,1]
                                 / length(ticker_data)), 4)
  q_ul <- round(wb_model$estimate[1] +
                    1.96 * sqrt( var$InvFisherInfMatrix[1,1]
                                 / length(ticker_data)), 4)
  res <- data.table(ticker, beta_ll, beta_ul, q_ll, q_ul)
  setnames(res, c("SYMBOL", "q_ll", "q_ul", "beta_ll", "beta_ul"))
  return(res)
}

s <- stocks
wb_gm_params <- rbindlist(lapply(s, discrete_fits))
write.csv(wb_gm_params, file="_dweibull_ztgeom_params_v1.csv")

wb_param_CI <- rbindlist(lapply(s, conf_int))
write.csv(wb_param_CI, file="_dweibull_param_CI_v1.csv")

################################################################################
################### Making Table for Thesis ####################
################################################################################

loglik_param <- as.data.table(read.csv("_dweibull_ztgeom_params_v1.csv"))
loglik_param <- loglik_param[ , .(w_loglik, g_loglik), by="SYMBOL"]
loglik_param <- loglik_param[, 2*(w_loglik - g_loglik), by="SYMBOL"]
write.csv(loglik_param, file="_loglik_param_v1.csv")

################################################################################
################################################################################
################### DWEIBULL FIT parameters vs securities ######################
################################################################################
```

```
387 ##############################################################################
388
389 dwb_params <- as.data.table(read.csv("_dweibull_ztgeom_params_v1.csv"))
390 dwb_params <- dwb_params[ , .(q, beta), by="SYMBOL"]
391 dwb_params[ , SECTOR := as.character(lapply(unlist(.SD), match_sector)),
392             .SDcols="SYMBOL"]
393 dwb_params[ , MARKETCAP := as.character(lapply(unlist(.SD), match_market_cap)),
394             .SDcols="SYMBOL"]
395
396 datum <- dwb_params[ stocks_stats, on = .(SYMBOL),
397                      ":=" ( N = N, V = V, mDTIME = mDTIME, mSIZE = mSIZE,
398                             q = q, beta = beta, SECTOR = SECTOR,
399                             MARKETCAP=MARKETCAP)]
400
401 ##############################################################################
402 ########################## FIT PARAM Scatter Plots ###########################
403 ##############################################################################
404
405 e1 <- ggplot(data=datum, aes(x=N, y=q)) +
406   geom_point(aes(color = factor(SECTOR)), size = 3) +
407   ggtitle(paste("Discrete Weibull Fit: Scale Parameter", sep="")) +
408   labs(x="N (Number of trades)", y="q") + ylim(0.2, 1)
409 ggsave(filename=paste("DW_scale_param_sec", ".pdf", sep=""), plot=e1)
410
411 e2 <- ggplot(data=datum, aes(x=N, y=q)) +
412   geom_point(aes(shape = factor(MARKETCAP)), size = 3) +
413   ggtitle(paste("Discrete Weibull Fit: Scale Parameter", sep="")) +
414   labs(x="N (Number of trades)", y="q") + ylim(0.2, 1)
415 ggsave(filename=paste("DW_scale_param_mcap", ".pdf", sep=""), plot=e2)
416
417 e3 <- ggplot(data=datum, aes(x=N, y=q)) +
418   geom_point(aes(color = factor(SECTOR), shape = factor(MARKETCAP)), size = 3) +
419   ggtitle(paste("Discrete Weibull Fit: Scale Parameter", sep="")) +
```

```
  labs(x="N (Number of trades)", y="q") + ylim(0.2, 1)
ggsave(filename=paste("DW_scale_param_mcap_sec", ".pdf", sep=""), plot=e3)

e4 <- ggplot(data=datum, aes(x=N, y=beta)) +
  geom_point(aes(color = factor(SECTOR)), size = 3) +
  ggtitle(paste("Discrete Weibull Fit: Shape Parameter", sep="")) +
  labs(x="N (Number of trades)", y="beta") + ylim(0.2, 1)
ggsave(filename=paste("DW_shape_param_sec", ".pdf", sep=""), plot=e4)

e5 <- ggplot(data=datum, aes(x=N, y=beta)) +
  geom_point(aes(shape = factor(MARKETCAP)), size = 3) +
  ggtitle(paste("Discrete Weibull Fit: Shape Parameter", sep="")) +
  labs(x="N (Number of trades)", y="beta") + ylim(0.2, 1)
ggsave(filename=paste("DW_shape_param_mcap", ".pdf", sep=""), plot=e5)

e6 <- ggplot(data=datum, aes(x=N, y=beta)) +
  geom_point(aes(color = factor(SECTOR), shape = factor(MARKETCAP)), size = 3) +
  ggtitle(paste("Discrete Weibull Fit: Shape Parameter", sep="")) +
  labs(x="N (Number of trades)", y="q") + ylim(0.2, 1)
ggsave(filename=paste("DW_shape_param_mcap_sec", ".pdf", sep=""), plot=e6)
```

time_dur_fit.R

B.5 orderbook_fit.R

```
################################################################################
##################### ORDER BOOK ACTIVITY: model and fits ######################
################################################################################
rm(list=ls())
library(highfrequency)
library(timeDate)
library(ggplot2)
```

```
library(data.table)
library(plyr)
library(fitdistrplus)
library(extraDistr)
library(stats)
library(CompGLM)
library(polyaAeppli)
library(gridExtra)
library(DiscreteWeibull)
library(actuar)
library(vcd)
theme_update(plot.title = element_text(hjust = 0.5))
datasource <- "~/Desktop/Research/RData"
setwd(datasource)

################################################################################
###################### Load data: AMZN stock at once ######################
################################################################################

stocks <- c( "AMZN" )
time_int_name <- function(x){
  return (paste("time_int_", x, ".RData", sep=""))
}
trd_data_name <- function(x){
  return (paste("trd_data_dt_", x, ".RData", sep=""))
}
qte_data_name <- function(x){
  return (paste("qte_data_dt_", x, ".RData", sep=""))
}
helper <- function(x){
  load(file=x)
  get(ls()[ls()!="filename"])
}
```

```
41
42 time_int_files <- unlist(lapply(stocks, time_int_name))
43 time_int_list <- lapply(time_int_files, helper)
44 time_int <- rbindlist(time_int_list)
45
46 trd_all_files <- unlist(lapply(stocks, trd_data_name))
47 trd_dt_list <- lapply(trd_all_files, helper)
48 trd_data_dt <- rbindlist(trd_dt_list)
49
50 qte_all_files <- unlist(lapply(stocks, qte_data_name))
51 qte_dt_list <- lapply(qte_all_files, helper)
52 qte_data_dt <- rbindlist(qte_dt_list)
53
54 ###############################################################################
55 ###################### Subset QUOTES data ON trade time ########################
56 ###############################################################################
57
58 days <- c(2:3,6:10,13:17,21:24,27:31) # trading days in the relevant month.
59
60 time_int <- time_int[ , ":=" (pSIZE = shift(SIZE,1), pPRICE = shift(PRICE,1),
61                               pDIR = shift(DIR,1)), by=.(DAY, SYMBOL)]
62 time_int <- na.omit(time_int)
63 time_int <- time_int[pSIZE==100 & DTIME>1, ] # Trades Size == 100 & DTIME>1
64
65 qte_data_dt_sub <- qte_data_dt
66 qte_data_dt_sub <- qte_data_dt_sub[ time_int,
67                                     on = .(DAY, TIME < tTIME, TIME > ptTIME),
68                                     ":=" ( ptTIME = ptTIME, DTIME = DTIME,
69                                            BID = BID-pPRICE, BIDSIZ = BIDSIZ,
70                                            OFR = OFR-pPRICE, OFRSIZ = OFRSIZ,
71                                            pDIR = pDIR, DIR = DIR, MOM = MOM)]
72 qte_data_dt_sub <- na.omit(qte_data_dt_sub)
73 qte_data_dt_sub <- qte_data_dt_sub[, TIME := TIME - ptTIME]
```

```
74
75 ticker <- stocks[1]
76 lob_act <- qte_data_dt_sub[ DTIME==10 & TIME%in%c(1:9), ]
77
78 lob_stats <- lob_act[ , .(mBID = mean(BID), sdBID = sd(BID),
79                           mBIDSIZ = mean(BIDSIZ), mOFR = mean(OFR),
80                           sdOFR = sd(OFR), mOFRSIZ = mean(OFRSIZ) ),
81                       by=.(pDIR, TIME) ][order(TIME)]
82
83 ##############################################################################
84 ##############################################################################
85 ######################### Histograms I: BID/ASK counts #########################
86 ##############################################################################
87 ##############################################################################
88
89 bin_w <- 0.1
90 a1 <- ggplot(lob_act[pDIR==1, ], aes(x=BID)) +
91   geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
92   geom_vline(xintercept = 0, color="black", size=1.5) +
93   ggtitle("BID side after BUY trade") + labs(x="rBID", y="Count") +
94   xlim(-2, 0.5) + ylim(0, 1600)
95
96 a2 <- ggplot(lob_act[pDIR==1, ], aes(x=OFR)) +
97   geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
98   geom_vline(xintercept = 0, color="black", size=1.5) +
99   ggtitle("ASK side after BUY trade") + labs(x="rASK", y="Count") +
100  xlim(-0.5, 2) + ylim(0, 1600)
101
102 a3 <- ggplot(lob_act[pDIR==-1, ], aes(x=BID)) +
103   geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
104   geom_vline(xintercept = 0, color="black", size=1.5) +
105   ggtitle("BID side after SELL trade") + labs(x="rBID", y="Count") +
106   xlim(-2, 0.5) + ylim(0, 1600)
```

```
107
108 a4 <- ggplot(lob_act[pDIR==-1, ], aes(x=OFR)) +
109   geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
110   geom_vline(xintercept = 0, color="black", size=1.5) +
111   ggtitle("ASK side after SELL trade") + labs(x="rASK", y="Count") +
112   xlim(-0.5, 2) + ylim(0, 1600)
113
114 plt <- grid.arrange( a1, a2, a3, a4, ncol = 2, nrow=2,
115                      top=paste(ticker, (": Aggregated LOB Activity"), sep=""))
116 ggsave(filename=paste("agg_lob_count_", ticker, ".pdf", sep=""), plot=plt)
117
118 ###############################################################################
119 ############################## Early time: t = 1s ##############################
120 ###############################################################################
121
122 b1 <- ggplot(lob_act[pDIR==1 & TIME==1, ], aes(x=BID)) +
123   geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
124   geom_vline(xintercept = 0, color="black", size=1.5) +
125   ggtitle("BID side after a BUY trade") + labs(x="rBID", y="Count") +
126   xlim(-2, 0.5) + ylim(0, 300)
127
128 b2 <- ggplot(lob_act[pDIR==1 & TIME==1, ], aes(x=OFR)) +
129   geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
130   geom_vline(xintercept = 0, color="black", size=1.5) +
131   ggtitle("ASK side after a BUY trade") + labs(x="rASK", y="Count") +
132   xlim(-0.5, 2) + ylim(0, 300)
133
134 b3 <- ggplot(lob_act[pDIR==-1 & TIME==1, ], aes(x=BID)) +
135   geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
136   geom_vline(xintercept = 0, color="black", size=1.5) +
137   ggtitle("BID side after a SELL trade") + labs(x="rBID", y="Count") +
138   xlim(-2, 0.5) + ylim(0, 300)
139
```

```
b4 <- ggplot(lob_act[pDIR==-1 & TIME==1, ], aes(x=OFR)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after a SELL trade") + labs(x="rASK", y="Count") +
  xlim(-0.5, 2) + ylim(0, 300)

plt <- grid.arrange( b1, b2, b3, b4, ncol = 2, nrow=2,
                     top = paste(ticker, (": early(t=1s) LOB Activity"), sep=""))
ggsave(filename=paste("early_lob_count_", ticker, ".pdf", sep=""), plot=plt)

################################################################################
############################## Late time: t = -1s ##############################
################################################################################

c1 <- ggplot(lob_act[pDIR==1 & TIME==9, ], aes(x=BID)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after a BUY trade") + labs(x="rBID", y="Count") +
  xlim(-2, 0.5) + ylim(0, 300)

c2 <- ggplot(lob_act[pDIR==1 & TIME==9, ], aes(x=OFR)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after a BUY trade") + labs(x="rASK", y="Count") +
  xlim(-0.5, 2) + ylim(0, 300)

c3 <- ggplot(lob_act[pDIR==-1 & TIME==9, ], aes(x=BID)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after a SELL trade") + labs(x="rBID", y="Count") +
  xlim(-2, 0.5) + ylim(0, 300)

c4 <- ggplot(lob_act[pDIR==-1 & TIME==9, ], aes(x=OFR)) +
```

```
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after a SELL trade") + labs(x="rASK", y="Count") +
  xlim(-0.5, 2) + ylim(0, 300)

plt <- grid.arrange( c1, c2, c3, c4, ncol = 2, nrow=2,
                     top=paste(ticker, (": late(t=9s) LOB Activity"), sep=""))
ggsave(filename=paste("late_lob_count_", ticker, ".pdf", sep=""), plot=plt)

################################################################################
################################################################################
######################### Histograms II: BID/ASK size ##########################
################################################################################
################################################################################

a1 <- ggplot(lob_act[pDIR==1, ], aes(x=BID, weight=BIDSIZ)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after BUY trade") + labs(x="rBID", y="bidSIZE") +
  xlim(-2, 0.5) + ylim(0, 3000)

a2 <- ggplot(lob_act[pDIR==1, ], aes(x=OFR, weight=OFRSIZ)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after BUY trade") + labs(x="rASK", y="askSIZE") +
  xlim(-0.5, 2) + ylim(0, 3000)

a3 <- ggplot(lob_act[pDIR==-1, ], aes(x=BID, weight=BIDSIZ)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after SELL trade") + labs(x="rBID", y="bidSIZE") +
  xlim(-2, 0.5) + ylim(0, 3000)

```

```
a4 <- ggplot(lob_act[pDIR==-1, ], aes(x=OFR, weight=OFRSIZ)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after SELL trade") + labs(x="rASK", y="askSIZE") +
  xlim(-0.5, 2) + ylim(0, 3000)

plt <- grid.arrange( a1, a2, a3, a4, ncol = 2, nrow=2,
                     top=paste(ticker, (": Aggregated LOB Activity"), sep=""))
ggsave(filename=paste("agg_lob_size_", ticker, ".pdf", sep=""), plot=plt)

################################################################################
############################## Early time: t = 1s ##############################
################################################################################

b1 <- ggplot(lob_act[pDIR==1 & TIME==1, ], aes(x=BID, weight=BIDSIZ)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after a BUY trade") + labs(x="rBID", y="bidSIZE") +
  xlim(-2, 0.5) + ylim(0, 500)

b2 <- ggplot(lob_act[pDIR==1 & TIME==1, ], aes(x=OFR, weight=OFRSIZ)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after a BUY trade") + labs(x="rASK", y="askSIZE") +
  xlim(-0.5, 2) + ylim(0, 500)

b3 <- ggplot(lob_act[pDIR==-1 & TIME==1, ], aes(x=BID, weight=BIDSIZ)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after a SELL trade") + labs(x="rBID", y="bidSIZE") +
  xlim(-2, 0.5) + ylim(0, 500)

b4 <- ggplot(lob_act[pDIR==-1 & TIME==1, ], aes(x=OFR, weight=OFRSIZ)) +
```

```
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after a SELL trade") + labs(x="rASK", y="askSIZE") +
  xlim(-0.5, 2) + ylim(0, 500)

plt <- grid.arrange( b1, b2, b3, b4, ncol = 2, nrow=2,
                     top=paste(ticker, (": early(t=1s) LOB Activity"), sep=""))
ggsave(filename=paste("early_lob_size_", ticker, ".pdf", sep=""), plot=plt)

################################################################################
############################## Late time: t = -1s ##############################
################################################################################

c1 <- ggplot(lob_act[pDIR==1 & TIME==9, ], aes(x=BID, weight=BIDSIZ)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after a BUY trade") + labs(x="rBID", y="bidSIZE") +
  xlim(-2, 0.5) + ylim(0, 500)

c2 <- ggplot(lob_act[pDIR==1 & TIME==9, ], aes(x=OFR, weight=OFRSIZ)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after a BUY trade") + labs(x="rASK", y="askSIZE") +
  xlim(-0.5, 2) + ylim(0, 500)

c3 <- ggplot(lob_act[pDIR==-1 & TIME==9, ], aes(x=BID, weight=BIDSIZ)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", boundary=bin_w/2) +
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("BID side after a SELL trade") + labs(x="rBID", y="bidSIZE") +
  xlim(-2, 0.5) + ylim(0, 500)

c4 <- ggplot(lob_act[pDIR==-1 & TIME==9, ], aes(x=OFR, weight=OFRSIZ)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", boundary=bin_w/2) +
```

```
  geom_vline(xintercept = 0, color="black", size=1.5) +
  ggtitle("ASK side after a SELL trade") + labs(x="rASK", y="askSIZE") +
  xlim(-0.5, 2) + ylim(0, 500)

plt <- grid.arrange( c1, c2, c3, c4, ncol = 2, nrow=2,
                     top=paste(ticker, (": late(t=9s) LOB Activity"), sep=""))
ggsave(filename=paste("late_lob_size_", ticker, ".pdf", sep=""), plot=plt)

################################################################################
################################ BINNING data ##################################
################################################################################
helper_cut <- function(x){
  # bin_cut <- seq(-1.05, 1.05, 0.1)
  # bin_val <- seq(-1, 1, 0.1)
  bin_cut <- seq(-2.05, 2.05, 0.1)
  bin_val <- seq(-2, 2, 0.1)
  f <- cut(x, bin_cut, labels=bin_val)
  return(as.numeric(levels(f))[f])
}
bid_cut <- function(x){
  # bin_cut <- seq(-1.05, 0.05, 0.1)
  # bin_val <- seq(-1, 0, 0.1)
  bin_cut <- seq(-2.05, 0.05, 0.1)
  bin_val <- seq(-2, 0, 0.1)
  f <- cut(x, bin_cut, labels=bin_val)
  return(as.numeric(levels(f))[f])
}
ask_cut <- function(x){
  # bin_cut <- seq(-0.05, 1.05, 0.1)
  # bin_val <- seq(0, 1, 0.1)
  bin_cut <- seq(-0.05, 2.05, 0.1)
  bin_val <- seq(0, 2, 0.1)
  f <- cut(x, bin_cut, labels=bin_val)
```

```
  return(as.numeric(levels(f))[f])
}

bid_side <- lob_act[ BID<=0, .(BIDSIZ), by=.(BID, pDIR, TIME) ]
bid_side[ , rP := lapply(.SD, bid_cut), .SDcols="BID"]
bid_side <- na.omit(bid_side)
bid_side <- bid_side[ , ":=" (N= .N, nSZ = sum(BIDSIZ)), by=.(rP, pDIR, TIME)]
bid_side <- bid_side[ , .(TIME, pDIR, rP, N, nSZ)][order(TIME, pDIR, rP)]
bid_side <- unique(bid_side, by=c("TIME", "pDIR", "rP"))

ask_side <- lob_act[ OFR>=0, .(OFRSIZ), by=.(OFR, pDIR, TIME)]
ask_side[ , rP := lapply(.SD, ask_cut), .SDcols="OFR"]
ask_side <- na.omit(ask_side)
ask_side <- ask_side[ , ":=" (N= .N, nSZ = sum(OFRSIZ)), by=.(rP, pDIR, TIME)]
ask_side <- ask_side[ , .(TIME, pDIR, rP, N, nSZ)][order(TIME, pDIR, rP)]
ask_side <- unique(ask_side, by=c("TIME", "pDIR", "rP"))

prep_bid_ask <- function(tau, bid_dt, ask_dt){
  bid_aB_ct <- bid_dt[TIME%in%tau & pDIR==1, rep(rP, N)]
  bid_aS_ct <- bid_dt[TIME%in%tau & pDIR==-1, rep(rP, N)]
  ask_aB_ct <- ask_dt[TIME%in%tau & pDIR==1, rep(rP, N)]
  ask_aS_ct <- ask_dt[TIME%in%tau & pDIR==-1, rep(rP, N)]
  bid_aB_sz <- bid_dt[TIME%in%tau & pDIR==1, rep(rP, nSZ)]
  bid_aS_sz <- bid_dt[TIME%in%tau & pDIR==-1, rep(rP, nSZ)]
  ask_aB_sz <- ask_dt[TIME%in%tau & pDIR==1, rep(rP, nSZ)]
  ask_aS_sz <- ask_dt[TIME%in%tau & pDIR==-1, rep(rP, nSZ)]
  bid_ask_dt <- rbind(bid_side[TIME%in%tau, ], ask_side[TIME%in%tau, ])
  return(list(bid_ask_dt, bid_aB_ct, bid_aS_ct, ask_aB_ct, ask_aS_ct,
              bid_aB_sz, bid_aS_sz, ask_aB_sz, ask_aS_sz))
}
tau <- c(9)
bid_ask_lst <- prep_bid_ask(tau, bid_side, ask_side)
bid_ask_dt <- bid_ask_lst[[1]]
```

```
bid_aB_ct <- bid_ask_lst[[2]]
bid_aS_ct <- bid_ask_lst[[3]]
ask_aB_ct <- bid_ask_lst[[4]]
ask_aS_ct <- bid_ask_lst[[5]]

################################################################################
######################### Distributions: BID/ASK side ##########################
################################################################################

bid_ct <- lob_act[ BID<=0, .(BIDSIZ), by=.(DAY, ptTIME, BID, pDIR, TIME) ]
bid_ct[ , rP := lapply(.SD, bid_cut), .SDcols="BID"]
bid_ct <- na.omit(bid_ct)
bid_ct <- bid_ct[ , N := .N, by=.(DAY, ptTIME, rP, pDIR, TIME)]
bid_ct <- bid_ct[ , .(DAY,
                      ptTIME, TIME, pDIR, rP, N)][order(DAY, ptTIME, TIME, pDIR, rP)]
bid_ct <- unique(bid_ct, by=c("DAY", "ptTIME", "TIME", "pDIR", "rP"))
ask_ct <- lob_act[ OFR>=0, .(OFRSIZ), by=.(DAY, ptTIME, OFR, pDIR, TIME) ]
ask_ct[ , rP := lapply(.SD, ask_cut), .SDcols="OFR"]
ask_ct <- na.omit(ask_ct)
ask_ct <- ask_ct[ , N := .N, by=.(DAY, ptTIME, rP, pDIR, TIME)]
ask_ct <- ask_ct[ , .(DAY,
                      ptTIME, TIME, pDIR, rP, N)][order(DAY, ptTIME, TIME, pDIR, rP)]
ask_ct <- unique(ask_ct, by=c("DAY", "ptTIME", "TIME", "pDIR", "rP"))

bid_sz <- lob_act[ BID<=0, .(BIDSIZ), by=.(DAY, ptTIME, BID, pDIR, TIME) ]
bid_sz[ , rP := lapply(.SD, bid_cut), .SDcols="BID"]
bid_sz <- na.omit(bid_sz)
bid_sz <- bid_sz[ , .(DAY,
                      ptTIME, TIME, pDIR, rP, BIDSIZ)][order(DAY, ptTIME, TIME, pDIR,
                          rP)]
ask_sz <- lob_act[ OFR>=0, .(OFRSIZ), by=.(DAY, ptTIME, OFR, pDIR, TIME) ]
ask_sz[ , rP := lapply(.SD, ask_cut), .SDcols="OFR"]
ask_sz <- na.omit(ask_sz)
```

```
ask_sz <- ask_sz[ , .(DAY,
                      ptTIME, TIME, pDIR, rP, OFRSIZ)][order(DAY, ptTIME, TIME, pDIR,
                          rP)]

bid_sum <- lob_act[ BID<=0, .(BIDSIZ), by=.(DAY, ptTIME, BID, pDIR, TIME) ]
bid_sum[ , rP := lapply(.SD, bid_cut), .SDcols="BID"]
bid_sum <- na.omit(bid_sum)
bid_sum <- bid_sum[ , S := sum(BIDSIZ), by=.(DAY, ptTIME, rP, pDIR, TIME)]
bid_sum <- bid_sum[ , .(DAY,
                        ptTIME, TIME, pDIR, rP, S)][order(DAY, ptTIME, TIME, pDIR, rP)]
bid_sum <- unique(bid_sum, by=c("DAY", "ptTIME", "TIME", "pDIR", "rP"))
ask_sum <- lob_act[ OFR>=0, .(OFRSIZ), by=.(DAY, ptTIME, OFR, pDIR, TIME) ]
ask_sum[ , rP := lapply(.SD, ask_cut), .SDcols="OFR"]
ask_sum <- na.omit(ask_sum)
ask_sum <- ask_sum[ , S := sum(OFRSIZ), by=.(DAY, ptTIME, rP, pDIR, TIME)]
ask_sum <- ask_sum[ , .(DAY,
                        ptTIME, TIME, pDIR, rP, S)][order(DAY, ptTIME, TIME, pDIR, rP)]
ask_sum <- unique(ask_sum, by=c("DAY", "ptTIME", "TIME", "pDIR", "rP"))

##### extract mean and var of bid/ask counts
zardoz <- function(t, pd){
  tmp1 <- bid_ct[TIME==t & pDIR==pd & rP==0, N]
  tmp2 <- ask_ct[TIME==t & pDIR==pd & rP==0, N]
  return(c(round(mean(tmp1),2), round(var(tmp1),2),
           round(mean(tmp2),2), round(var(tmp2),2)))
}
zardoz(1,1)

##### check poisson hypothesis
tt <- 1
pd <- 1
bid_sd <- bid_ct[TIME==tt & pDIR==pd & rP==0, N]
ask_sd <- ask_ct[TIME==tt & pDIR==pd & rP==0, N]
```

```
402
403 gf1 <- goodfit(bid_sd, type="poisson", method = "MinChisq")
404 gf2 <- goodfit(ask_sd, type="poisson", method = "MinChisq")
405 summary(gf1)
406 summary(gf2)
407
408 bin_w <- 1
409 tt <- c(1)
410 dP <- c(0)
411
412 c1 <- ggplot(bid_ct[TIME%in%tt & pDIR==1 & rP%in%dP, ], aes(x=N)) +
413   geom_histogram(binwidth=bin_w, fill="blue", color="black", center=bin_w) +
414   ggtitle( "BID side after a BUY trade") + labs(x="N (events)", y="Counts") +
415   xlim(0, 15) + ylim(0, 80)
416 c2 <- ggplot(ask_ct[TIME%in%tt & pDIR==1 & rP%in%dP, ], aes(x=N)) +
417   geom_histogram(binwidth=bin_w, fill="red", color="black", center=bin_w) +
418   ggtitle( "ASK side after a BUY trade") + labs(x="N (events)", y="Counts") +
419   xlim(0, 15) + ylim(0, 80)
420 c3 <- ggplot(bid_ct[TIME%in%tt & pDIR==-1 & rP%in%dP, ], aes(x=N)) +
421   geom_histogram(binwidth=bin_w, fill="blue", color="black", center=bin_w) +
422   ggtitle( "BID side after a SELL trade") + labs(x="N (events)", y="Counts") +
423   xlim(0, 15) + ylim(0, 80)
424 c4 <- ggplot(ask_ct[TIME%in%tt & pDIR==-1 & rP%in%dP, ], aes(x=N)) +
425   geom_histogram(binwidth=bin_w, fill="red", color="black", center=bin_w) +
426   ggtitle( "ASK side after a SELL trade") + labs(x="N (events)", y="Counts") +
427   xlim(0, 15) + ylim(0, 80)
428 plt <- grid.arrange( c1, c2, c3, c4, ncol = 2, nrow=2,
429                      top=paste(ticker, (": Distribution of LOB Activity
430                                         at |rP|=0, t=1s"), sep=""))
431 ggsave(filename=paste("lob_act_ct_rp0_t1_", ticker, ".pdf", sep=""), plot=plt)
432
433
434 d1 <- ggplot(bid_sz[TIME%in%tt & pDIR==1 & rP%in%dP, ], aes(x=BIDSIZ)) +
```

```
  geom_histogram(binwidth=bin_w, fill="blue", color="black", center=bin_w) +
  ggtitle( "BID side after a BUY trade") + labs(x="BIDSIZ (events)", y="Counts") +
  xlim(0, 15) + ylim(0, 300)
d2 <- ggplot(ask_sz[TIME%in%tt & pDIR==1 & rP%in%dP, ], aes(x=OFRSIZ)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", center=bin_w) +
  ggtitle( "ASK side after a BUY trade") + labs(x="ASKSIZ (events)", y="Counts") +
  xlim(0, 15) + ylim(0, 300)
d3 <- ggplot(bid_sz[TIME%in%tt & pDIR==-1 & rP%in%dP, ], aes(x=BIDSIZ)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black", center=bin_w) +
  ggtitle( "BID side after a SELL trade") + labs(x="BIDSIZ (events)", y="Counts") +
  xlim(0, 15) + ylim(0, 300)
d4 <- ggplot(ask_sz[TIME%in%tt & pDIR==-1 & rP%in%dP, ], aes(x=OFRSIZ)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black", center=bin_w) +
  ggtitle( "ASK side after a SELL trade") + labs(x="ASKSIZ (events)", y="Counts") +
  xlim(0, 15) + ylim(0, 300)
plt <- grid.arrange( d1, d2, d3, d4, ncol = 2, nrow=2,
                     top=paste(ticker, (": Distribution of LOB Activity
                                         at rP=0, t=1s"), sep=""))
ggsave(filename=paste("lob_act_sz_rp0_t1_", ticker, ".pdf", sep=""), plot=plt)


e1 <- ggplot(bid_sum[TIME%in%tt & pDIR==1 & rP%in%dP, ], aes(x=S)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black",
                 center=bin_w, aes(y=..density..)) +
  ggtitle( "BID side after BUY") + labs(x="BIDSIZ (events)", y="Density") +
  xlim(0, 20) + ylim(0, 1)
e2 <- ggplot(ask_sum[TIME%in%tt & pDIR==1 & rP%in%dP, ], aes(x=S)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black",
                 center=bin_w, aes(y=..density..)) +
  ggtitle( "ASK side after BUY") + labs(x="ASKSIZ (events)", y="Density") +
  xlim(0, 20) + ylim(0, 1)
e3 <- ggplot(bid_sum[TIME%in%tt & pDIR==-1 & rP%in%dP, ], aes(x=S)) +
  geom_histogram(binwidth=bin_w, fill="blue", color="black",
```

```
                center=bin_w, aes(y=..density..)) +
  ggtitle( "BID side after SELL") + labs(x="BIDSIZ (events)", y="Density") +
  xlim(0, 20) + ylim(0, 1)
e4 <- ggplot(ask_sum[TIME%in%tt & pDIR==-1 & rP%in%dP, ], aes(x=S)) +
  geom_histogram(binwidth=bin_w, fill="red", color="black",
                center=bin_w, aes(y=..density..)) +
  ggtitle( "ASK side after SELL") + labs(x="ASKSIZ (events)", y="Density") +
  xlim(0, 20) + ylim(0, 1)
plt <- grid.arrange( e1, e2, e3, e4, ncol = 2, nrow=2,
                    top=paste(ticker, (": Dist Density of LOB Activity
                                       at rP=0, t=1s"), sep=""))
ggsave(filename=paste("lob_act_sumsz_rp0_t1_", ticker, ".pdf", sep=""), plot=plt)
```

orderbook_fit.R

BIBLIOGRAPHY

[1] Michael Aitken, Amaryllis Kua, Philip Brown, Terry Watter, and H Y. Izan. An intraday analysis of the probability of trading on the asx at the asking price. *Australian Journal of Management*, 20(2):115–154, 1995.

[2] Aurélien Alfonsi, Antje Fruth, and Alexander Schied. Optimal execution strategies in limit order books with general shape functions. *Quantitative Finance*, 10(2):143–157, 2010.

[3] William Vickers Anthony Bagnall, Jason Lines and Eamonn Keogh. The uea & ucr time series classification repository. `http://www.timeseriesclassification.com`.

[4] Louis Bachelier. *Louis Bachelier's theory of speculation: the origins of modern finance*. Princeton University Press, 2011.

[5] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 31(3):606–660, 2017.

[6] Aurelio Fernandez Bariviera, Luciano Zunino, M Belén Guercio, Lisana B Martinez, and Osvaldo A Rosso. Efficiency and credit ratings: a permutation-information-theory analysis. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(08):P08007, 2013.

[7] Dimitris Bertsimas and Andrew W Lo. Optimal control of execution costs. *Journal of Financial Markets*, 1(1):1–50, 1998.

[8] Bruno Biais, Pierre Hillion, and Chester Spatt. An empirical analysis of the limit order book and the order flow in the paris bourse. *The Journal of Finance*, 50(5):1655–1689, 1995.

[9] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of political economy*, 81(3):637–654, 1973.

[10] Adam Blazejewski and Richard Coggins. A local non-parametric model for trade sign inference. *Physica A: Statistical Mechanics and its Applications*, 348:481–495, 2005.

[11] Jean-Philippe Bouchaud. Price impact. *Encyclopedia of quantitative finance*, 2010.

[12] Jean-Philippe Bouchaud, Yuval Gefen, Marc Potters, and Matthieu Wyart. Fluctuations and response in financial markets: the subtle nature of ?random?price changes. *Quantitative finance*, 4(2):176–190, 2004.

[13] Jean-Philippe Bouchaud, Julien Kockelkoren, and Marc Potters. Random walks, liquidity molasses and critical response in financial markets. *Quantitative finance*, 6(02):115–123, 2006.

[14] Jean-Philippe Bouchaud, Marc Mzard, and Marc Potters. Statistical properties of stock order books: empirical results and models. *Quantitative Finance*, 2(4):251–256, 2002.

[15] Thomas Bury. A statistical physics perspective on criticality in financial markets. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(11):P11004, 2013.

[16] Anirban Chakraborti, Ioane Muni Toke, Marco Patriarca, and Frederic Abergel. Econophysics review: I. empirical facts. *Quantitative Finance*, 11(7):991–1012, 2011.

[17] Damien Challet and Robin Stinchcombe. Analyzing and modeling 1+1d markets. *Physica A: Statistical Mechanics and its Applications*, 300(1):285 – 299, 2001.

[18] Rama Cont. Empirical properties of asset returns: stylized facts and statistical issues. 2001.

[19] Rama Cont. Statistical modeling of high-frequency financial data. *IEEE Signal Processing Magazine*, 28:16–25, 2011.

[20] Rama Cont and Adrien De Larrard. Order book dynamics in liquid markets: limit theorems and diffusion approximations. 2012.

[21] Rama Cont and Adrien De Larrard. Price dynamics in a markovian limit order market. *SIAM Journal on Financial Mathematics*, 4(1):1–25, 2013.

[22] Rama Cont, Arseniy Kukanov, and Sasha Stoikov. The price impact of order book events. *Journal of financial econometrics*, 12(1):47–88, 2014.

[23] Rama Cont, Sasha Stoikov, and Rishi Talreja. A stochastic model for order book dynamics. *Operations research*, 58(3):549–563, 2010.

[24] Harald Cramér. *Mathematical Methods of Statistics (PMS-9)*, volume 9. Princeton university press, 2016.

[25] Z. Eisler and J. Kertész. Size matters: some stylized facts of the stock market revisited. *The European Physical Journal B - Condensed Matter and Complex Systems*, 51(1):145–154, May 2006.

[26] Tristan Fletcher, Zakria Hussain, and John Shawe-Taylor. Multiple kernel learning on the limit order book. In *Proceedings of the First Workshop on Applications of Pattern Analysis*, pages 167–174, 2010.

[27] A Ronald Gallant. *Nonlinear statistical models*, volume 310. John Wiley & Sons, 2009.

[28] Martin D Gould, Mason A Porter, Stacy Williams, Mark McDonald, Daniel J Fenn, and Sam D Howison. Limit order books. *Quantitative Finance*, 13(11):1709–1742, 2013.

[29] Olivier Guéant, Charles-Albert Lehalle, and Joaquin Fernandez-Tapia. Optimal portfolio liquidation with limit orders. *SIAM Journal on Financial Mathematics*, 3(1):740–764, 2012.

[30] Plamen Ch. Ivanov, Ainslie Yuen, Boris Podobnik, and Youngki Lee. Common scaling patterns in intertrade times of u. s. stocks. *Phys. Rev. E*, 69:056107, May 2004.

[31] P. Jain. Institutional Design and Liquidity at Stock Exchanges Around the World by Pankaj Jain :: SSRN. *Available on SSRN*, 2003.

[32] Franck Jovanovic and Christophe Schinckus. Breaking down the barriers between econophysics and financial economics. *International Review of Financial Analysis*, 47(Supplement C):256 – 266, 2016.

[33] Alec N Kercheval and Yuan Zhang. Modelling high-frequency limit order book dynamics with support vector machines. *Quantitative Finance*, 15(8):1315–1329, 2015.

[34] D. O. Ledenyov and V. O. Ledenyov. On the Risk Management with Ap-

plication of Econophysics Analysis in Central Banks and Financial Institutions. *ArXiv e-prints*, November 2012.

[35] CHARLES M. C. LEE and MARK J. READY. Inferring trade direction from intraday data. *The Journal of Finance*, 46(2):733–746, 1991.

[36] Rosario N Mantegna and H Eugene Stanley. *Introduction to econophysics: correlations and complexity in finance*. Cambridge university press, 1999.

[37] Rosario Nunzio Mantegna. Presentation of the english translation of ettore majorana's paper: The value of statistical laws in physics and social sciences. *Quantitative Finance*, 5(2):133–140, 2005.

[38] Rosario Nunzio Mantegna. The tenth article of ettore majorana. *Europhysics News*, 37(4):15–17, 2006.

[39] Yuriy Nevmyvaka, Yi Feng, and Michael Kearns. Reinforcement learning for optimized trade execution. In *Proceedings of the 23rd international conference on Machine learning*, pages 673–680. ACM, 2006.

[40] Anna A Obizhaeva and Jiang Wang. Optimal trading strategy and supply/demand dynamics. *Journal of Financial Markets*, 16(1):1–32, 2013.

[41] Mauro Politi and Enrico Scalas. Fitting the empirical distribution of intertrade durations. *Physica A: Statistical Mechanics and its Applications*, 387(8):2025 – 2034, 2008.

[42] Calyampudi Radhakrishna Rao, Calyampudi Radhakrishna Rao, Mathematischer Statistiker, Calyampudi Radhakrishna Rao, and Calyampudi Radhakrishna Rao. *Linear statistical inference and its applications*, volume 2. Wiley New York, 1973.

[43] Peter Richmond, Jürgen Mimkes, and Stefan Hutzler. *Econophysics and physical economics*. Oxford University Press, 2013.

[44] Dean Rickles. Econophysics for philosophers. *Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics*, 38(4):948 – 978, 2007.

[45] Ioanid Roşu. A dynamic model of the limit order book. *The Review of Financial Studies*, 22(11):4601–4641, 2009.

[46] Gheorghe Savoiu. *Econophysics: Background and Applications in Economics, Finance, and Sociophysics*. Academic Press, 2013.

[47] Enrico Scalas, Rudolf Gorenflo, Hugh Luckock, Francesco Mainardi, Maurizio Mantelli, and Marco Raberto. Anomalous waiting times in high-frequency financial data. *Quantitative Finance*, 4(6):695–702, 2004.

[48] Christophe Schinckus. Is econophysics a new discipline? the neopositivist argument. *Physica A: Statistical Mechanics and its Applications*, 389(18):3814 – 3821, 2010.

[49] Kimani A. Stancil. Introduction to statistical mechanics; statistical mechanics in a nutshell. *Physics Today*, 65, 2012.

[50] Haider Syed and Amar K Das. Identifying chemotherapy regimens in electronic health record data using interval-encoded sequence alignment. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 143–147. Springer, 2015.

[51] Haider Syed and Amar K Das. Temporal needleman-wunsch. In *Data Science and Advanced Analytics (DSAA), 2015. 36678 2015. IEEE International Conference on*, pages 1–9. IEEE, 2015.

[52] Hideki Takayasu. *Empirical science of financial fluctuations: The advent of econophysics.* Springer Science & Business Media, 2013.

[53] Peter Tino, Nikolay Nikolaev, and Xin Yao. Volatility forecasting with sparse bayesian kernel models. In *Proc. 4th International Conference on Computational Intelligence in Economics and Finance, Salt Lake City, UT*, pages 1150–1153, 2005.

[54] Johannes Voit. *The statistical mechanics of financial markets*. Springer Science & Business Media, 2013.

[55] Gang-Jin Wang and Chi Xie. Cross-correlations between wti crude oil market and u.s. stock market: A perspective from econophysics. 43:2021–2036, 10 2012.

[56] Luciano Zunino, Massimiliano Zanin, Benjamin M. Tabak, Daro G. Prez, and Osvaldo A. Rosso. Complexity-entropy causality plane: A useful approach to quantify the stock market inefficiency. *Physica A: Statistical Mechanics and its Applications*, 389(9):1891–1901, 2010.